

OS/390



Infoprint Server Customization

OS/390



Infoprint Server Customization

Note

Before using this information and the product it supports, be sure to read the general information in "Notices" on page 147.

Second Edition (December 1999)

This edition applies to OS/390 Version 2 Release 8 Modification Level 0, Program Number 5647-A01, and to all subsequent releases and modifications until otherwise indicated in new editions or technical newsletters. Be sure to use the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

The IBM Printing Systems Company welcomes your comments. A form for reader's comments is provided at the back of this publication. If the form has been removed, you may send your comments to the following address:

INFORMATION DEVELOPMENT
THE IBM PRINTING SYSTEMS COMPANY
DEPARTMENT H7FE BUILDING 003G
BOULDER CO 80301-9191
USA

If you prefer to send comments electronically, use one of the following methods:

- Internet: printpub@us.ibm.com
- Fax: 1-800-524-1519 or 1-303-924-6873

Internet

Visit our home pages at <http://www.ibm.com/printers> and <http://www.ibm.com/s390/os390/>

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 1995 1999. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Figures	v	Customizing Print Interface Support for the OS/390 SMB Server	37
Tables	vii	Customizing the Print Interface IPP Server	38
About This Publication	ix	Editing the aopd.conf Configuration File	38
Who Should Use This Publication	ix	Setting Environment Variables	39
How This Publication is Organized	ix	APF-Authorizing Libraries	39
Related Information.	ix	Customizing the IPP Workstation Client.	39
Understanding Syntax Notation	x		
Summary of Changes	xiii		
Chapter 1. Introducing Infoprint Server	1	Chapter 5. Customizing Infoprint Server Transforms and the Transform Manager 43	
Printer Inventory Manager.	3	Enabling Infoprint Server Transforms.	44
Windows Client	4	Creating the aopxfd.conf Configuration File	44
Print Interface	4	PCL to AFP Transform Classes	44
Infoprint Server Transforms for OS/390 and the Transform Manager	7	Format	45
NetSpool	8	Example	47
IP PrintWay	10	Editing the aopd.conf Configuration File	48
SNMP Subagent	12	Setting Environment Variables	49
Chapter 2. Customization Roadmap	15	Installing Multibyte Conversion Tables for SAP to AFP Transform	49
Chapter 3. Customizing the Printer Inventory Manager.	17	Customizing SAP to AFP Configuration Files	50
Enabling the Infoprint Server Feature.	17	barcode.tab Configuration File	50
Making Run-time Libraries Available	18	defcp.tab Configuration File	52
Creating HFS Directories	18	fonts.tab Configuration File	54
Creating the Infoprint Server Configuration File	19	image.tab Configuration File.	56
Setting Environment Variables	20	pagedef.tab Configuration File	57
Establishing Security for the Printer Inventory and Operator Commands	21	xxxx0000.tab Configuration File.	59
Creating Startup and Shutdown Procedures	22	Chapter 6. Customizing the SNMP Subagent.	61
AOPSTART Startup Procedure	22	Customizing and Starting the OS/390 SNMP Agent	61
AOPSTOP Shutdown Procedure	23	Editing the aopd.conf Configuration File	62
Enabling Infoprint Server ISPF Panels	24	Enabling SNMP Reporting in the Printer Inventory	62
Defining Libraries in the TSO Logon Procedure	24	Customizing the SNMP Manager	63
Displaying the Infoprint Server Panel Option	24	Chapter 7. Customizing NetSpool	65
Modifying the AOPINIT EXEC	25	Allocating and Initializing the Message Log	65
Backing up the Printer Inventory	25	Creating a NetSpool Startup Procedure	66
Chapter 4. Customizing Print Interface	27	Determining the Number of NetSpool Programs to Start	66
General Customization for Print Interface	27	Using Japanese or Spanish National Language Versions	66
Customizing TCP/IP	28	Using an Extended MCS Console	66
Customizing JES.	29	NetSpool Startup Procedure	67
Editing the aopd.conf Configuration File	29	JCL Parameters	67
Writing a Filter Program	30	Writing NetSpool Exit Programs	69
Customizing for UNIX Printing Commands	34	Programming Considerations	69
Customizing the AOPPRINT JCL Procedure	35	APIPP1X2 Control Block	70
Customizing the Print Interface LPD	35	Beginning of File Exit (APIPPTD1).	72
Editing the aopd.conf Configuration File	36	Transparent Data Control Exit (APIPPTD2).	73
Customizing the LPR	36	Sample Exits and Macro	75
		Installing the Exits	76
		Modifying the Exits	76
		Chapter 8. Customizing IP PrintWay	77

Customizing TCP/IP	77
Customizing TCPIP.DATA Data Set	77
Customizing the PROFILE.TCPIP Data Set	78
Customizing the IPP Client	78
Defining the IP PrintWay Functional Subsystem	79
Determining How Many Functional Subsystems to Define	79
Determining How Many Functional Subsystem Applications to Define.	79
Determining Work-Selection Criteria	80
Defining the IP PrintWay Functional Subsystem in JES2	80
Defining the IP PrintWay Functional Subsystem in JES3	83
APF-Authorizing SYS1.IMAGELIB.	86
Allocating and Initializing the Transmission-Queue Data Set	86
Restricting Access to the Transmission-Queue Data Set	87
Allocating and Initializing the IP PrintWay Message-Log Data Set	87
Creating an IP PrintWay Startup Procedure.	88
Startup Procedure	88
JCL Statements	89
Writing IP PrintWay Exit programs	90
Programming Considerations	92
Common Control Blocks	92
Routing Exit (ANFUXRTG)	96
Begin Data Set Exit	102
Record Exit	106
End Data Set Exit	111
SMF Exit (ANFUXSMF)	115
Message Exit (ANFUXMSG)	117
Response Notification Exit (ANFUXRSP)	119
Using the Block Letter Program (ANFUBLK)	122
Sample Exits	124
Installing Exits	124
Modifying Exits	125
Customizing PSF for OS/390 for Sharing Network Printers	125

Chapter 9. Installing and Customizing the Windows Client. 127

Installing and Customizing the OS/390 Printer Port Monitor	127
Installing the AFP Printer Driver	128
Installing the AFP Viewer Plug-In	128

Appendix A. Infoprint Server Files . . 131

aopd.conf.	131
Purpose	131
Format	131
Usage Notes.	133
Related Files.	134
aopxfd.conf	135
Purpose	135
Format	135
Usage Notes.	138
Examples.	138
Related Files.	139

Appendix B. Environment Variables 141

Appendix C. SCS Printer Simulation 143

Appendix D. 3270 Data Streams . . . 145

Command Codes	145
Control Codes	145
Attribute Types.	146

Notices 147

Programming Interfaces	148
Trademarks	148

Glossary 151

Sources	151
References	151

Bibliography. 161

Infoprint Server for OS/390	161
Print Services Facility for OS/390.	161
Advanced Function Presentation (AFP).	161
OS/390 Version 2 Release 8.	161
OS/390 SecureWay Communications Server	162
CICS for OS/390	163
IMS/ESA® Release 5	163
3270 Data Stream	163

Index 165

Readers' Comments — We'd Like to Hear from You 171

Figures

1.	Infoprint Server Components	1	16.	Heisei Gothic Portion of Default fonts.tab	
2.	Printer Inventory Manager	3		Configuration File	56
3.	Print Interface System Diagram	5	17.	Default image.tab Configuration File	57
4.	Transform Manager and Infoprint Server		18.	Default pagedef.tab Configuration File	59
	Transforms for OS/390	7	19.	Default 80000000.tab Configuration File	60
5.	NetSpool System Diagram	9	20.	NetSpool Startup Procedure —	
6.	IP PrintWay System Diagram.	11		SYS1.PROCLIB(APIPJCL).	67
7.	SNMP Subagent System Diagram	13	21.	Format of APIPP1X2 Control Block.	71
8.	AOPSTART Startup Procedure —		22.	Sample JES2 Initialization Statements	80
	SYS1.PROCLIB(AOPSTART)	23	23.	Sample JES3 Initialization Statements	83
9.	AOPSTOP Shutdown Procedure —		24.	IDCAMS Parameters to Allocate a	
	SYS1.PROCLIB(AOPSTOP)	24		Transmission-Queue Data Set	87
10.	Sample UNIX Filter	32	25.	IP PrintWay Startup Procedure —	
11.	AOPPRINT JCL Procedure in SYS1.PROCLIB	35		SYS1.PROCLIB(ANFWPROC)	89
12.	Default barcode.tab Configuration File	52	26.	Format of the IP PrintWay Exit Parameter Area	
13.	Default defcp.tab Configuration File	53		(ANFUEXTTP)	94
14.	Sample defcp.tab.japan Configuration File	54			
15.	Header and Courier Portions of Default				
	fonts.tab Configuration File	56			

Tables

1.	Summary of Infoprint Server Publications	x		4.	SAP Code Page Configuration Files	60
2.	Summary of Infoprint Server Functions and Corresponding Infoprint Server Components .	15		5.	Comparison of Printing Methods	127
3.	IPP Attributes and Corresponding Infoprint Server Attributes.	40		6.	SCS Code Points	143

About This Publication

This publication provides information about how to customize the following products:

- Infoprint® Server for OS/390®, hereafter called Infoprint Server
- Infoprint Server Transforms for OS/390 (5697-F51), a separate IBM® licensed product, hereafter called Infoprint Server Transforms

Who Should Use This Publication

This publication is intended for system programmers and administrators responsible for customizing Infoprint Server for their installation. The reader should be familiar with OS/390 UNIX® System Services, TCP/IP, the Job entry subsystem (JES), and OS/390 job control language (JCL).

How This Publication is Organized

This publication is organized so that you can read it sequentially or refer directly to individual tasks.

Chapter 1 introduces Infoprint Server and describes how the different components of Infoprint Server fit into your system and the functions each component provides. Use this introduction to determine which components of Infoprint Server you want to use in your installation. This chapter is identical in content to the introduction chapter found in other Infoprint Server publications (*OS/390 Infoprint Server Operation and Administration* and *OS/390 Infoprint Server Messages and Diagnosis*); therefore, if you have read the introduction in other publications, you do not need to read it again.

Chapter 2 lists the major functions provided by Infoprint Server, identifies the components of Infoprint Server that you need to customize, and directs you to the sections in this publication that describe the customization tasks you need to perform.

Chapters 3 through 9 describe the customization tasks required to customize each component of Infoprint Server. Read only the chapters that pertain to the component you want to customize for your installation. Except for the Printer Inventory Manager component, which you must customize in order to use any Infoprint Server function, you can customize components at different times, as your installation decides to use the function.

The appendices provide the following reference information:

- Summary of the syntax of Infoprint Server configuration files
- Summary of the environment variables used by Infoprint Server

A bibliography lists all Infoprint Server publications and other related publications.

Related Information

“Bibliography” on page 161 lists the publications referred to in this book and publications that contain additional information about related products.

For additional information about Infoprint Server, visit this Web site:
<http://www.ibm.com/printers>

For additional information about OS/390 and for the latest OS/390 publications and documentation updates that result from authorized program analysis reports (APARs) and program temporary fixes (PTFs), visit this Web site:
<http://www.ibm.com/s390/os390>

Table 1. Summary of Infoprint Server Publications

Publication	Form number
<i>OS/390 Infoprint Server Introduction</i> Introduces all components of Infoprint Server, including IP PrintWay™, NetSpool™, and Print Interface. Contains printing scenarios that show how you can use Infoprint Server in your installation.	G544-5696
<i>OS/390 Infoprint Server Migration</i> Summarizes the new function in Infoprint Server for OS/390 V2R8, describes required and optional migration tasks to implement the new function in your installation, and describes the Infoprint Server migration program, which helps the administrator convert IP PrintWay, NetSpool, and Print Interface printer information to the format required by Infoprint Server for OS/390 V2R8.	G544-5697
<i>OS/390 Infoprint Server Customization</i> Describes customization tasks for all components of Infoprint Server, including IP PrintWay, NetSpool, and Print Interface. Describes required environment variables, configuration files, and startup procedures.	G544-5694
<i>OS/390 Infoprint Server Operation and Administration</i> Describes operator procedures and administrative tasks for all components of Infoprint Server, including IP PrintWay, NetSpool, and Print Interface. Describes in detail how to create entries and specify printer attributes in the Printer Inventory.	S544-5693
<i>OS/390 Infoprint Server User's Guide</i> Describes how to submit print jobs from remote systems (including Windows® systems), the local OS/390 system, and Virtual Telecommunications Access Method (VTAM®) applications. Describes in detail the lp , lpstat , cancel , pcl2afp , ps2afp , pdf2afp , and sap2afp UNIX commands; the AOPPRINT JCL procedure; OUTPUT JCL parameters supported by IP PrintWay; and the Infoprint Server Windows client.	S544-5692
<i>OS/390 Infoprint Server Messages and Diagnosis</i> Describes messages issued by all components of Infoprint Server, including IP PrintWay, NetSpool, and Print Interface. Also describes how to diagnose and report errors.	G544-5690

Understanding Syntax Notation

The following rules apply to coding illustrations throughout this publication:

- Uppercase or bold letters are to be coded as shown.
- Variable data is printed in italics. Enter specific data to replace the characters in italics.

- Do not enter the following symbols as part of a parameter or option unless directed to do so:

Vertical Bar |

Underscore ____

Brackets []

Braces { }

- A vertical bar between two values means that you select one of the values.
- An underscored value means that if an option is not specified, the underscored value, called the default, is used.
- Brackets around a value mean that you do not have to select the value.
- Braces around a value generally mean that you must select one of the values; however, in some cases, you must code the braces. The text identifies when the braces are required.

Summary of Changes

Summary of Changes for *OS/390 Infoprint Server Customization*, G544-5694-01

This edition contains the following *new* information:

- “Customizing Print Interface Support for the OS/390 SMB Server” on page 37 describes some considerations for customizing the OS/390 Server Message Block (SMB) server for printing to Print Interface from Windows applications that use SMB protocol.
- “Installing Multibyte Conversion Tables for SAP to AFP Transform” on page 49 and “Customizing SAP to AFP Configuration Files” on page 50 describe how to customize the SAP to AFP transform that is provided with Infoprint Server Transforms.

This edition also contains the following *changed* information:

- “Enabling the Infoprint Server Feature” on page 17 and “Enabling Infoprint Server Transforms” on page 44 contains information about enabling the Infoprint Server feature and the Infoprint Server Transforms product on your OS/390 system.
- “Writing a DLL Filter” on page 31 describes enhancements for installation-written DLL filters. Print Interface passes additional information to a DLL filter, and a DLL filter can modify job attributes.
- “Customizing the IPP Workstation Client” on page 39 lists the IPP attributes that the Print Interface IPP server supports.
- “Chapter 6. Customizing the SNMP Subagent” on page 61 describes the need to recycle Network Printer Manager (NPM) for the Web when you enable or disable SNMP reporting for a PSF printer after initially defining the OS/390 print system to NPM.

Technical changes and additions are indicated by a vertical line to the left of the changes.

Summary of Changes for *OS/390 Infoprint Server Customization*, G544-5694-00

This publication contains the customization information previously presented in the following publications:

- *OS/390 Print Interface Configuration Guide*, G544-5544-01
- *IBM NetSpool Guide*, G544-5301-05
- *IBM IP PrintWay Guide*, S544-5379-04

This publication contains technical additions and changes related to enhancements made in Infoprint Server for OS/390 Version 2 Release 8 (V2R8). Refer to *OS/390 Infoprint Server Migration* for a summary of the V2R8 enhancements.

Chapter 1. Introducing Infoprint Server

Infoprint Server for OS/390, an optional element of OS/390 Version 2 Release 8, provides support for LAN and host printing on OS/390. Infoprint Server consists of several components that work together to provide printing services. Figure 1 shows some of the components and how they fit into your system. The components and features of Infoprint Server are shaded in the figure. Following the figure is a description of each Infoprint Server component.

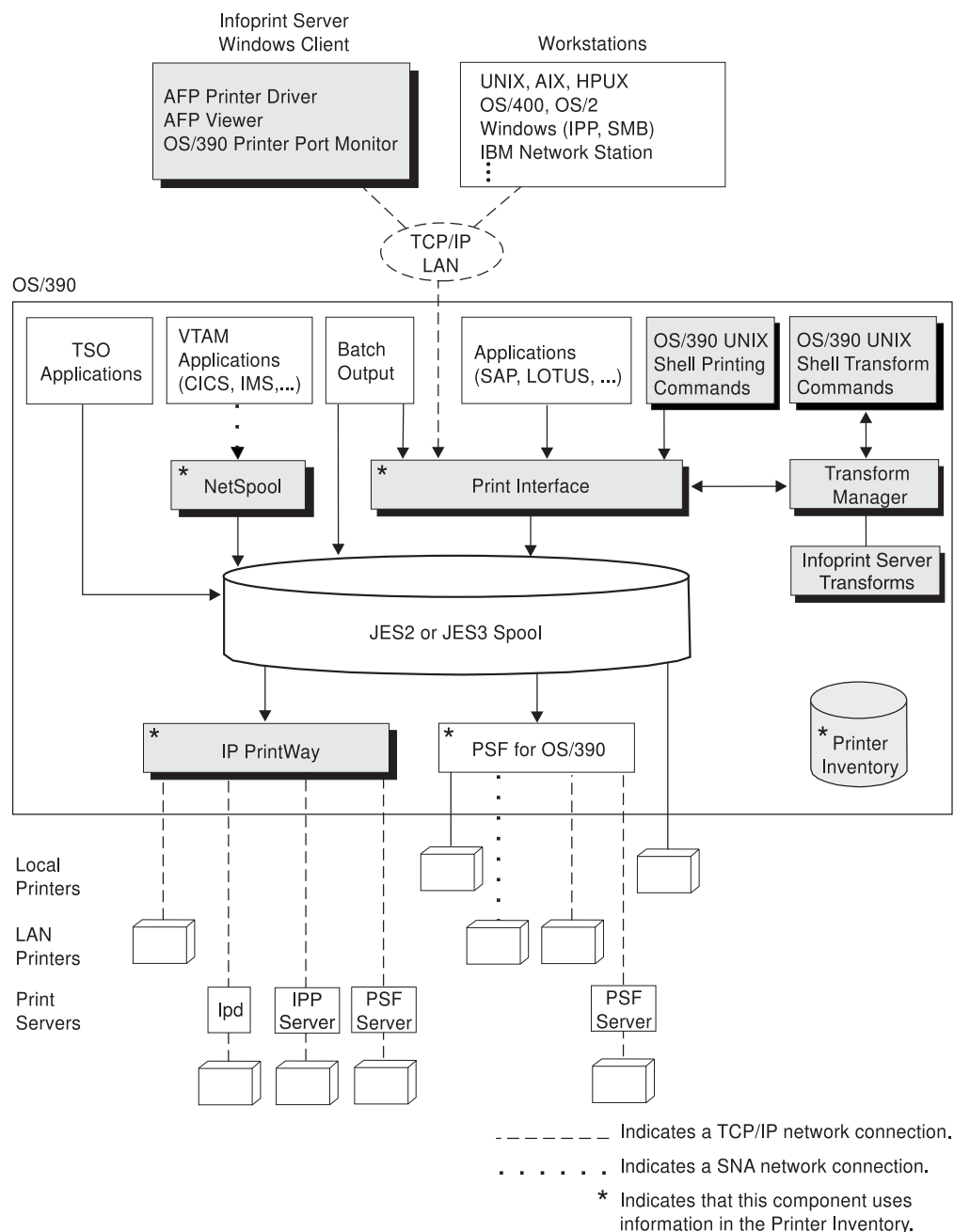


Figure 1. Infoprint Server Components

Printer Inventory and Printer Inventory Manager

The Printer Inventory Manager controls the Printer Inventory, a set of files in the hierarchical file system (HFS) that contain information about each printer to which NetSpool, Print Interface, and IP PrintWay can print. The Printer Inventory also contains system configuration information for IP PrintWay and, optionally, for PSF for OS/390.

Infoprint Server Windows client

The Windows client provides (1) an AFP™ printer driver, (2) an AFP viewer plug-in, and (3) an OS/390 printer port monitor that sends print requests to the Print Interface component.

Print Interface

Print Interface processes print requests from remote clients that use any of the following TCP/IP printing protocols:

- Line printer requester (LPR) to line printer daemon (LPD)
- Internet Printing Protocol (IPP)
- Server Message Block (SMB) protocol

Print Interface also provides OS/390 UNIX shell commands (**lp**, **lpstat**, and **cancel**) and the AOPPRINT JCL procedure to let local users submit print requests to Print Interface.

Print Interface accepts any data format the target printer can print, converts data to EBCDIC or ASCII as required by the target printer, and allocates output data sets on the JES spool. Print Interface can also transform PCL, PDF, PostScript, and SAP data to AFP format prior to writing data to the JES spool, for printing on IBM AFP printers.

Infoprint Server Transforms for OS/390 and the Transform Manager

Infoprint Server Transforms for OS/390 is a licensed program product (5697-F51) that provides PCL, PostScript, PDF, and SAP to AFP transforms for the OS/390 system. The Transform Manager component of Infoprint Server manages the PCL, PostScript, and PDF transforms.

Infoprint Server Transforms also provides OS/390 UNIX shell commands (**pcl2afp**, **ps2afp** and its alias **pdf2afp**, and **sap2afp**) to let local users transform data without printing it.

NetSpool

NetSpool processes print requests from VTAM applications, such as CICS® and IMS™. NetSpool accepts SCS, 3270, and binary data and allocates output data sets on the JES spool.

IP PrintWay

IP PrintWay transmits data sets from the JES spool to printers or print servers using any of the following TCP/IP protocols:

- Line printer requester (LPR) to line printer daemon (LPD)
- Internet Printing Protocol (IPP)
- Direct socket printing

Simple Network Management Protocol (SNMP) subagent (not shown in figure)

The SNMP subagent lets you use an SNMP manager to view printer characteristics and printer status for printers controlled by PSF for OS/390 that do not have internal SNMP agents or are not TCP/IP-attached to PSF.

Printer Inventory Manager

The Printer Inventory Manager controls the Printer Inventory, HFS files that contain information about the OS/390 printing environment. The administrator must create and manage information in the Printer Inventory.

The administrator can create the following objects:

- Printer definitions, which contain information about printers to which Print Interface, NetSpool, or IP PrintWay can print.
- Printer pool definitions, which contain information about groups of printers to which NetSpool can broadcast data.
- FSA definitions, which contain configuration information for IP PrintWay and PSF for OS/390 functional subsystem applications (FSAs).
- FSS definitions, which contain configuration information for IP PrintWay and PSF for OS/390 functional subsystems (FSSs).

Figure 2 shows how the administrator can create definitions in the Printer Inventory and which components of Infoprint Server use the Printer Inventory.

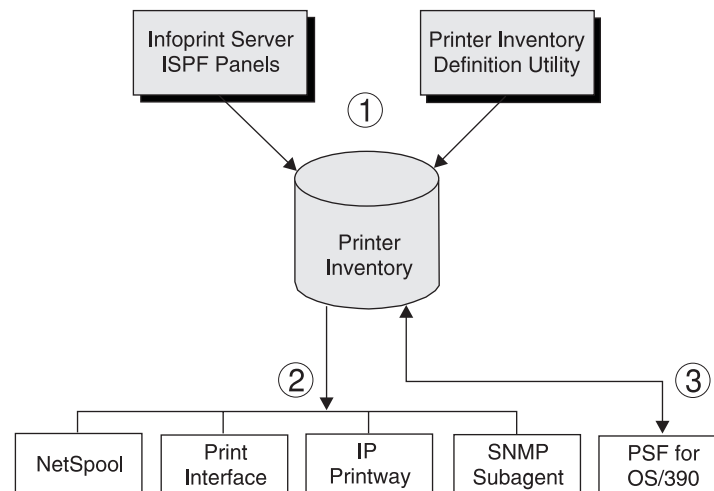


Figure 2. Printer Inventory Manager

1. The administrator can use Infoprint Server ISPF panels and the Printer Inventory Definition Utility (PIDU) to create and maintain the Printer Inventory. The PIDU is useful for creating many printer definitions at the same time and for backing up the Printer Inventory.
2. The following Infoprint Server components use information in the Printer Inventory:
 - NetSpool uses information in printer definitions and in printer pool definitions.
 - Print Interface uses information in printer definitions.
 - IP PrintWay uses information in printer definitions. IP PrintWay also can use IP PrintWay configuration information in FSS and FSA definitions.
 - The SNMP subagent uses printer information that PSF for OS/390 stores in the Printer Inventory about PSF printers.
3. PSF for OS/390, although not a component of Infoprint Server, can use configuration information that the administrator specifies in FSS and FSA definitions. PSF for OS/390 can also store printer information in the Printer

Inventory for use by the Infoprint Server SNMP subagent. For information about how to customize PSF for OS/390 to use the Printer Inventory, refer to *PSF for OS/390: Customization*.

Additional functions provided by the Printer Inventory Manager are:

- **Migration program**

The Infoprint Server migration program helps the administrator migrate from previous releases of IP PrintWay, NetSpool, and the OS/390 Print Server. The migration program merges printer information currently specified in NetSpool print characteristics data sets, NetSpool tables, NetSpool startup procedures, IP PrintWay routing and options data sets, and the Print Interface printer inventory to create entries (such as printer definitions and printer pool definitions) in the new Infoprint Server Printer Inventory.

The migration program can also move printer information in PSF startup procedures to FSS and FSA definitions in the Printer Inventory.

- **Security**

The administrator can restrict access to the Printer Inventory and to the operator commands that start and stop the Printer Inventory Manager, the Print Interface LPD, the Print Interface IPP server, the Transform Manager, and the SNMP subagent.

Windows Client

The Infoprint Server Windows client consists of the following programs:

AFP Printer Driver for Windows

The AFP Printer Driver creates output files in AFP format, so that users can print documents to IBM AFP printers. The AFP Printer Driver can create output files containing documents, overlays, or page segments. It can also create inline form definitions for printing documents with special options, such as printing on both sides of the paper.

AFP Viewer Plug-in for Windows

The AFP Viewer plug-in lets users view documents in AFP format, for example documents downloaded from the OS/390 system or documents on the Web. The AFP Viewer plug-in also lets users print AFP documents to AFP as well as non-AFP printers.

OS/390 Printer Port Monitor for Windows

The OS/390 Printer Port Monitor lets users print documents using standard print-submission methods from any Windows application that supports printing. After the OS/390 Printer Port Monitor is installed and configured on the Windows system, the Port Monitor automatically sends documents to the Print Interface component of Infoprint Server.

Note: Infoprint Server also supports printing from a Windows system using the SMB protocol. To use the SMB protocol, Windows users do not need to install the Printer Port Monitor.

Print Interface

Print Interface accepts print requests from remote clients and local users. Figure 3 on page 5 shows the steps that occur from the time Print Interface receives a print request until it allocates an output data set on the JES spool. An explanation of each step follows.

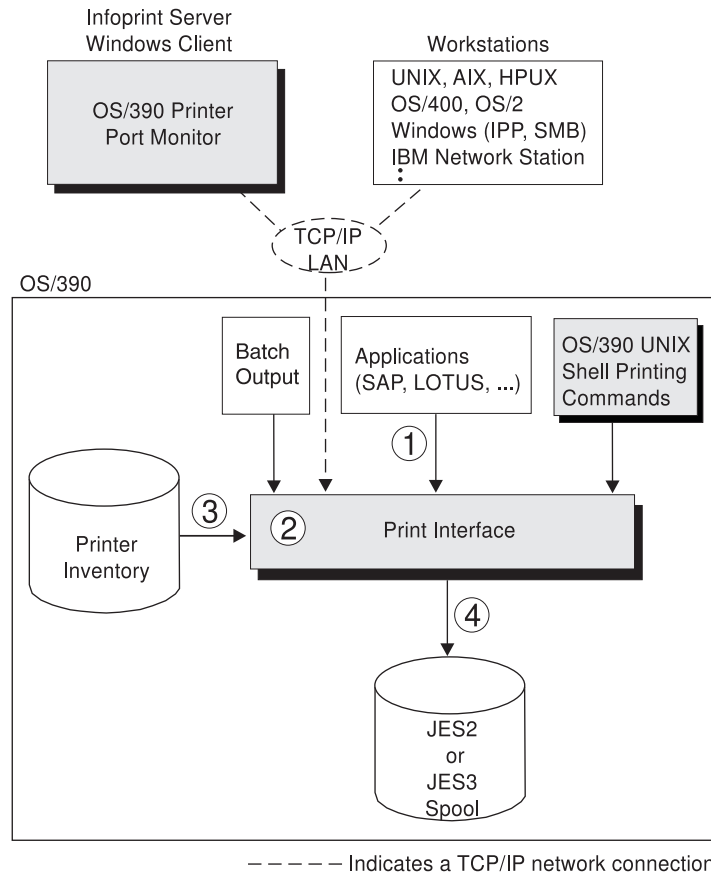


Figure 3. Print Interface System Diagram

1. Users can submit print requests from remote clients that use one of the following TCP/IP protocols:
 - LPR to LPD. The OS/390 Printer Port Monitor for Windows and commands, such as **lpr** and **lpq**, use this protocol.
 - Internet Printing Protocol (IPP).
 - Server Message Block (SMB) printing protocol.

Users can submit print requests from the local system with one of the following methods:

- OS/390 UNIX shell printing commands (**lp**, **lpstat**, and **cancel**). These commands, which adhere to the XPG4.2 standard, let users print HFS files and traditional data sets, query the status of a print job, and cancel a print job.
 - The AOPPRINT JCL procedure, which lets users print HFS files and traditional data sets.
2. Print Interface runs as a UNIX application that uses the services of OS/390 UNIX System Services.

Print Interface accepts data in any format, including but not limited to the following formats: S/390[®] line data, MO:DCA-P (also known as AFP), PostScript, PDF, PCL, SAP, and text. Print Interface automatically detects the data format and can validate that the printer accepts that data format. It can also convert data to EBCDIC or ASCII, and transform data from one format to another.

3. Each print request specifies the name of a printer definition for the target printer in the Printer Inventory. Print Interface uses information in the printer definition to determine how to process the data, whether or not to transform the data, and so on.
4. Print Interface dynamically allocates output data sets on the JES2 or JES3 spool using JES allocation parameters specified in the printer definition, including:
 - JES work-selection parameters, such as class, forms name, and destination. These parameters cause JES to direct the output data sets to the correct JES output writer or functional subsystem application (FSA), such as PSF for OS/390 or IP PrintWay.
 - Advanced Function Presentation (AFP) parameters, such as the name of a form definition and page definition. PSF for OS/390 uses these parameters when printing data on IBM AFP printers.

Some additional functions provided by Print Interface include:

- **Validation of print requests**

Before accepting print requests, Print Interface can validate, with some exceptions, that the document can print as requested on the selected printer. For example, Print Interface can reject documents with data formats that the printer does not support.

- **Data Transforms**

Print Interface can transform data if requested in the printer definition in the Printer Inventory:

- Print Interface can transform S/390 line data (for example, in a sequential data set or a partitioned data set) into text data for printing on a printer such as an IBM network printer.
- Print Interface can transform text data into S/390 line data for printing on an IBM AFP printer.
- Print Interface can use Infoprint Server Transforms to transform data to AFP format.
- Print Interface can transform PCL, PDF, and PostScript data to AFP format remotely on an AIX[®] system running Infoprint Manager for AIX V2R1 or higher; and can transform PCL and PostScript data to AFP format on an AIX system running PSF for AIX V2R1.

- **Notification of completion**

Print Interface can notify users on the local OS/390 system when processing of a document is complete and the data set has been removed from the JES spool. It can also notify users who request mail notification with a command, such as **lpr**, that uses the LPR to LPD protocol.

- **Status reporting**

Print Interface can report the status of its data sets that are still on the JES spool. It can report if the data set has been selected for processing, held by the system, retained due to a failed transmission to a LAN printer, or deleted before printing.

1. Print Interface uses the transform filters provided by Infoprint Server Transforms to transform data on the OS/390 system. The administrator must configure the printer definitions to use the transform filters; by default, Print Interface does not transform data. The administrator and users can also specify transform filter options that control the transforms.

As an alternative to performing the transforms on the OS/390 system, Print Interface can transform PCL, PostScript, and PDF data to AFP format remotely on an AIX system running Infoprint Manager for AIX (or PSF for AIX). The Infoprint Server Transforms product does not need to be installed for remote transforms.
2. Users can use OS/390 UNIX shell commands (**pcl2afp**, **ps2afp** and its alias **pdf2afp**, and **sap2afp**), provided with Infoprint Server Transforms, to transform data without printing it.
3. The Transform Manager manages the PCL and PostScript/PDF transforms and controls how many transform daemons are active at one time. The Transform Manager does not manage the SAP to AFP transform, because the SAP to AFP transform is not implemented as a daemon.
4. Infoprint Server Transforms provides PCL to AFP, PostScript/PDF to AFP, and SAP to AFP transforms.

NetSpool

NetSpool intercepts print data from VTAM applications, such as CICS and IMS, converts the data into S/390 line data, and creates output data sets on the JES2 or JES3 spool. You can configure NetSpool so that you do not need to change existing VTAM applications; that is, existing VTAM applications can send print requests to NetSpool in the same manner as they currently send print requests to SNA network printers.

Figure 5 on page 9 shows the steps that occur from the time VTAM applications send print requests to NetSpool printer logical units (LUs) until NetSpool allocates output data sets on the JES spool. An explanation of each step follows.

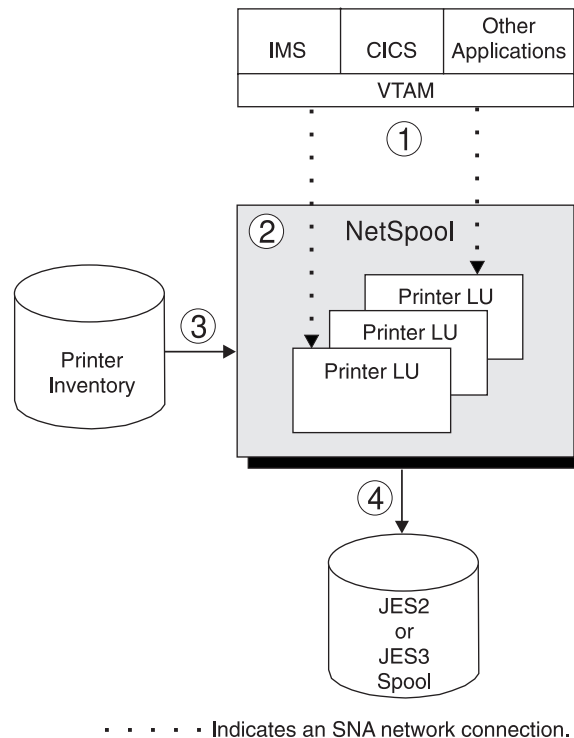


Figure 5. NetSpool System Diagram

1. VTAM applications, such as CICS or IMS, establish communication sessions with NetSpool printer logical units (LUs) instead of with SNA-network printers. Each NetSpool printer LU must be defined to VTAM as an application logical-unit (LU).
NetSpool can process the following types of VTAM data streams:
 - SNA character string (SCS) data over an LU type 1 session
 - 3270 data over an LU type 3 or LU type 0 session
 - A binary data stream over an LU type 0, type 1, or type 3 session
2. NetSpool runs as a VTAM application on the same or different OS/390 system. Multiple instances of NetSpool can run simultaneously in separate address spaces; each instance of NetSpool can process VTAM print requests sent to different NetSpool printer LUs.
3. Each NetSpool printer LU must be defined in a printer definition in the Printer Inventory. NetSpool converts the data stream into S/390 line-data format and groups the data into output data sets using information in the printer definition.
4. NetSpool dynamically allocates output data sets on the JES2 or JES3 spool using JES allocation parameters specified in the printer definition, including:
 - JES work-selection parameters, such as class, forms name, and destination. These parameters cause JES to direct the output data sets to the correct JES output writer or functional subsystem application (FSA), such as PSF for OS/390 or IP PrintWay.
 - Advanced Function Presentation (AFP) parameters, such as the name of a form definition and page definition. PSF for OS/390 uses these parameters when printing data on IBM AFP printers.
 - Distribution information, such as name and address, which can be printed on output header pages

NetSpool can also broadcast the same data to several different printers. To broadcast data, NetSpool allocates multiple output data sets on the JES spool.

Additional functions provided by NetSpool are:

- **Operator control**

The system operator can control NetSpool processing from the system console and from extended MCS consoles by issuing NetSpool commands while NetSpool is running. The operator can start and stop individual printer LUs and display the status of printer LUs. To assist in managing data sets from the console, the names of the output data sets created by NetSpool identify the VTAM application that generated the print request.

- **SCS data stream support**

NetSpool converts an SCS data stream into variable-length records, each starting with an ASA carriage-control character. Refer to the appendices in *OS/390 Infoprint Server Customization* for information about how NetSpool supports SCS code points and commands.

- **Transparent data support for SCS data**

NetSpool supports the Transparent (TRN) control in SCS data, which identifies the start of a transparent data stream. NetSpool provides two installation exits that let you add transparent data to the beginning of a data set and modify or delete transparent data whenever it occurs in the data stream.

- **3270 data stream support**

NetSpool converts a 3270 data stream into variable-length print records, each starting with an ASA carriage-control character. Refer to the appendices in *OS/390 Infoprint Server Customization* for information about how NetSpool supports code points for the 3270 data stream.

- **Binary data support**

The administrator can request in the printer definition that NetSpool treat the data stream as binary data. NetSpool writes binary data to the output data set as variable length records without formatting the data and without rejecting unsupported commands, orders, or data. This function is useful if you want to pass through all data without change and without including transparent (TRN) controls.

IP PrintWay

IP PrintWay transmits output data sets from the JES spool to printers or print servers in a TCP/IP network using one of the following TCP/IP protocols: LPR to LPD, IPP, or direct-socket printing.

Figure 6 on page 11 shows the steps that occur from the time IP PrintWay selects output data sets from the JES spool until IP PrintWay transmits the data sets to the target printer or print server and deletes the data sets from the JES spool. An explanation of each step follows.

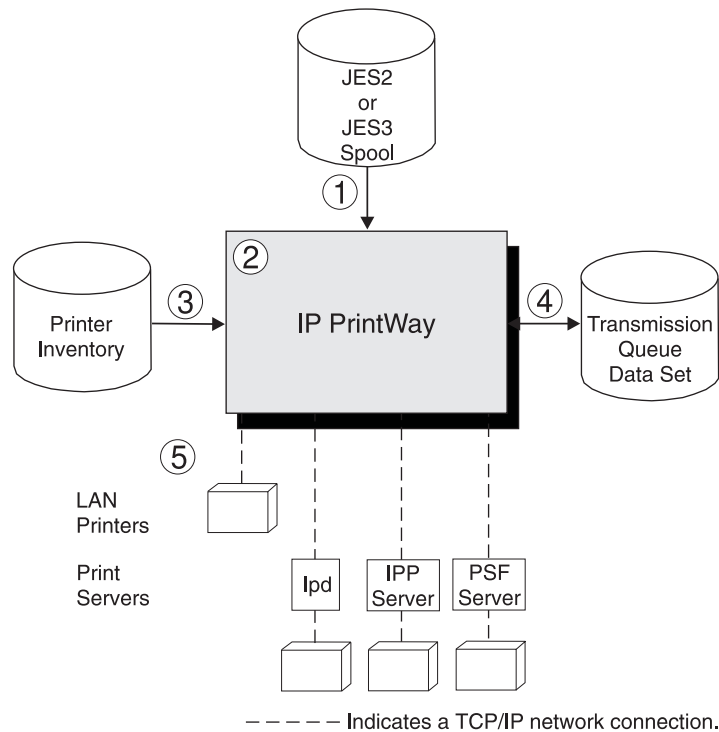


Figure 6. IP PrintWay System Diagram

1. IP PrintWay selects output data sets from the JES spool according to the JES work-selection parameters defined for the IP PrintWay FSA. For example, IP PrintWay might select all data sets in JES output class P.

IP PrintWay can select data sets that were allocated on the JES spool by NetSpool or Print Interface, or submitted from TSO or batch applications. The data sets can contain S/390 line data, ASCII text data, or formatted data, such as PCL, PostScript, SAP, or MO:DCA-P (AFP) data.

2. IP PrintWay runs as a functional subsystem application (FSA) of JES2 or JES3. Several IP PrintWay FSAs can run in one functional subsystem address space (FSS) to handle a high volume of data; however, one PrintWay FSA can transmit data sets to multiple printers or print servers.
3. IP PrintWay uses information in the printer definition in the Printer Inventory to process data sets, select the TCP/IP transmission protocol (LPR, IPP, or direct sockets), and obtain the address of the target printer. IP PrintWay can also use the IP address of a target printer specified directly on the OUTPUT JCL statement.

IP PrintWay recognizes data sets allocated on the JES spool by Print Interface and does not convert data from ASCII to EBCDIC or format the data; this is because Print Interface has already converted data to ASCII if necessary. For other data sets, IP PrintWay can convert data from EBCDIC to ASCII, can add a header to each page, and can format data using the carriage-control characters in S/390 line data, an FCB, or pagination attributes specified in the printer definition.

4. IP PrintWay maintains a transmission queue to keep track of data sets being processed. This transmission queue contains the status of each transmission, routing information, and so on. Using Infoprint Server ISPF panels, the system operator can monitor the status of transmissions, reroute data sets to another print queue or port, and change the transmission options.

5. IP PrintWay can use the LPR to LPD, IPP, or direct-socket TCP/IP protocol to transmit data sets to remote printers or print servers. IP PrintWay also transmits LPD options and IPP job attributes to the target LPDs and IPP servers. For example, IP PrintWay can transmit information for the LPD to print on a separator page.

IP PrintWay retries an unsuccessful transmission automatically for a short period of time right after transmission. In addition, IP PrintWay can also retry an unsuccessful transmission for a specified number of times at a specified interval. Retry limits and retry times can be specified in the printer definition or on an OUTPUT JCL statement.

Additional functions provided by IP PrintWay include:

- **Retaining jobs on the JES spool**

After successfully transmitting each data set, or after completing the requested number of transmission attempts, IP PrintWay can retain the data set on the JES spool forever or for a period of time. Retention periods can be specified in the printer definition or on an OUTPUT JCL statement.

- **Printer selection using an OUTPUT JCL statement**

On an OUTPUT JCL statement, a user can select the printer definition by specifying either (1) the name of the printer definition in the FSSDATA parameter or (2) the DEST, CLASS, or FORMS parameter (or a combination of these parameters) associated with the printer definition.

Users can also specify the IP address for the target printer directly on the OUTPUT JCL statement, thereby eliminating the need for the administrator to create a printer definition for each printer in the Printer Inventory.

- **Accounting**

For each data set processed, IP PrintWay writes a System Management Facility (SMF) type-6 record, which includes the number of bytes transmitted and the IP address of the target system.

- **Installation exits**

IP PrintWay supports exits written by an installation to customize IP PrintWay processing. For example, an exit can change the IP address of the remote printer, add separator pages, modify SMF accounting records, and notify users of processing events.

- **Maintaining transmission order**

IP PrintWay preserves the order of the data sets on the JES spool when transmitting data sets. IP PrintWay retains this order even if the transmission of the data sets must be retried. If a job contains more than one output data set, IP PrintWay acquires all of the data sets in the job before transmitting any of them and can transmit the data sets to the remote printer as a single file. Although the data sets are a single file, each data set starts printing on a new page.

- **Transmitting printer instructions**

The administrator can specify printer instructions in the printer definition for IP PrintWay to send to the printer before or after the data to be printed. Printer instructions might change fonts or switch between simplex and duplex printing.

SNMP Subagent

The Infoprint Server SNMP subagent, in conjunction with support provided by PSF for OS/390 and the OS/390 SNMP agent, lets administrators monitor printer characteristics (such as the printer resolution) and printer status (such as paper jams) for any printer controlled by PSF for OS/390. Also, administrators can be

notified as soon as an intervention situation (such as a paper jam) occurs on the printer. This support does not let administrators change any printer characteristics.

To monitor PSF printers, the OS/390 SNMP agent must be configured on the OS/390 system and an SNMP manager, such as IBM Network Printer Manager (NPM) for the Web, must be installed.

You can monitor PSF printers that do not contain internal SNMP agents, such as the IBM 3900 printer, and also PSF printers that have internal SNMP agents but are not TCP/IP-attached to PSF. You can also monitor PSF TCP/IP-attached printers that contain internal SNMP agents; however, consider defining PSF printers that have internal SNMP agents directly to the SNMP manager. When you define a printer directly to the SNMP manager, you can also monitor printer statistics and change some printer characteristics. Refer to the documentation for your printers to determine if they have internal SNMP agents.

Figure 7 shows how the SNMP subagent fits into your system. An explanation of each step follows:

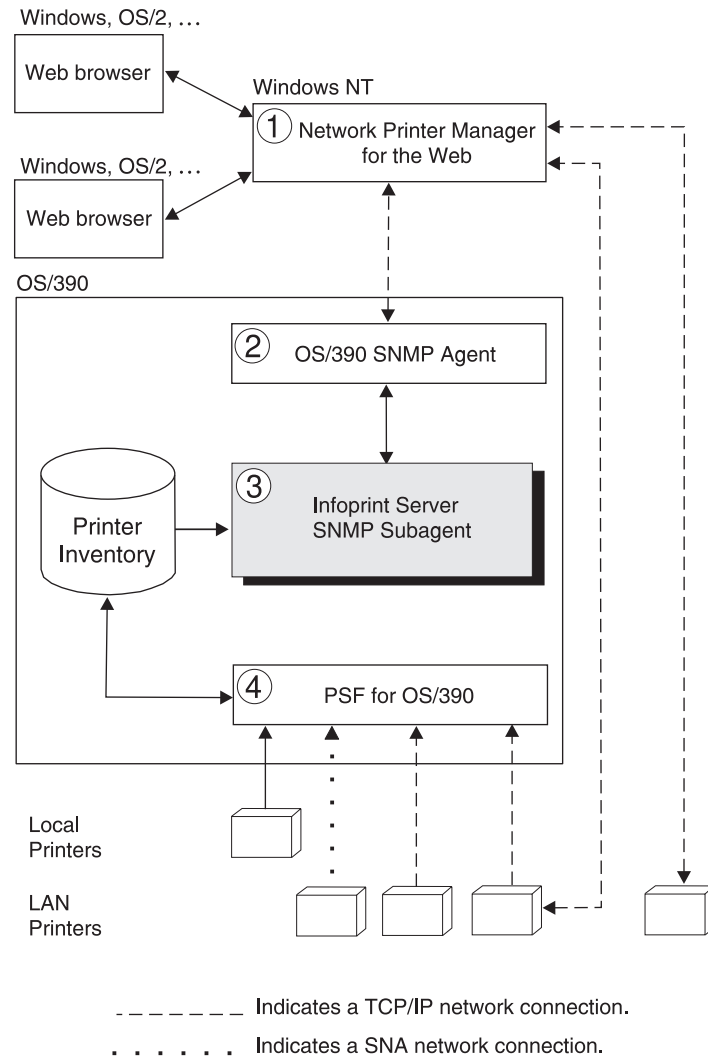


Figure 7. SNMP Subagent System Diagram

1. To monitor PSF printers, you must install an SNMP manager. You can install any SNMP manager that communicates with an SNMP agent that supports the general printer MIB (defined in RFC 1759) and has implemented support for multiple printers defined with one IP address.

One SNMP manager that provides the required support is IBM Network Printer Manager (NPM) for the Web, which is shown in Figure 7 on page 13. You can download NPM from the IBM Printing Systems Company Web site. Administrators can use a Java[®]-enabled version of Netscape Navigator or Microsoft[®] Internet Explorer to monitor PSF printers, while the NPM server runs on a Windows NT[®] system. Note that NPM limits the number of PSF printers an administrator can monitor at the same time; see the NPM online help for the maximum number of printers NPM lets you monitor.

As shown in the figure, the SNMP manager (NPM) can also communicate directly with any TCP/IP-attached printer that contains an internal SNMP agent; this printer can be controlled by PSF, but this is not necessary.

2. The OS/390 SNMP agent, part of the OS/390 SecureWay[®] Communications Server, communicates with the SNMP manager and with the Infoprint Server SNMP subagent.
3. The Infoprint Server SNMP subagent communicates with PSF for OS/390 through the Printer Inventory. PSF for OS/390 stores printer characteristics and printer status in the Printer Inventory for any printer that has SNMP-reporting enabled. (The administrator enables SNMP reporting in the FSA definition for the PSF printer).

The SNMP subagent transmits the information stored by PSF for OS/390 to the SNMP manager through the OS/390 SNMP agent. The SNMP subagent also notifies the SNMP manager immediately when PSF for OS/390 detects a change to printer characteristics or printer status.

4. PSF for OS/390 obtains printer characteristics and printer status from any PSF-controlled printer. The PSF printer can be channel-attached, TCP/IP-attached, or SNA-attached; however, the PSF printer cannot be attached using the Download for OS/390 feature of PSF.

If a PSF-controlled printer is shared with another printing application, and is not connected to PSF when an intervention required situation occurs, PSF cannot report the change in printer status.

Chapter 2. Customization Roadmap

This chapter can help you determine which Infoprint Server components you must customize to use the different functions that Infoprint Server provides. Table 2 lists the functions provided by Infoprint Server, the Infoprint Server components that you need to customize, and the sections in this publication that describe the required customization tasks.

Note that you must customize the Printer Inventory Manager component in order to use most functions provided by Infoprint Server. You can customize other components at a later date as your installation decides to use additional functions provided by Infoprint Server.

If you plan to use more than one Infoprint Server function, customize all related components. For example, to print from VTAM applications to TCP/IP-attached printers, customize the Printer Inventory Manager, NetSpool, and IP PrintWay.

Table 2. Summary of Infoprint Server Functions and Corresponding Infoprint Server Components

Infoprint Server Function	Components	See Page:
Receive print requests from the following sources, optionally transform data, and allocate output data sets on the JES spool: <ul style="list-style-type: none">• Clients that use LPR to LPD protocol• Clients that use Internet Printing Protocol (IPP)• Clients that use Server Message Block (SMB) protocol• OS/390 UNIX lp, lpstat, and cancel commands• AOPPRINT JCL procedure• Windows applications that support printing	Printer Inventory Manager	17
	Print Interface	27
	Infoprint Server Transforms and Transform Manager (to transform data on OS/390)	43
	OS/390 Printer Port Monitor for Windows (optional)	127
Receive print requests from VTAM applications (such as CICS and IMS) and allocate output data sets on the JES spool.	Printer Inventory Manager	17
	NetSpool	65
Transform data with the pcl2afp , ps2afp , pdf2afp and sap2afp commands.	Printer Inventory Manager	17
	Infoprint Server Transforms and Transform Manager	43
Transmit output data sets from the JES spool using one of the following TCP/IP protocols: <ul style="list-style-type: none">• LPR to LPD protocol• Internet Printing Protocol (IPP)• Direct sockets printing	Printer Inventory Manager	17
	IP PrintWay	77
View printer characteristics and status of PSF printers.	Printer Inventory Manager	17
	SNMP subagent	61
Store PSF for OS/390 system information in the Printer Inventory.	Printer Inventory Manager	17
View AFP documents from a Windows system.	AFP Viewer plug-in	127
Transform documents to AFP format on a Windows system.	AFP Printer Driver	127

Chapter 3. Customizing the Printer Inventory Manager

The Printer Inventory Manager component of Infoprint Server maintains the Printer Inventory and provides Infoprint Server ISPF panels and the Printer Inventory Definition Utility program (PIDU) for the administrator to create and manage entries in the Printer Inventory. See “Printer Inventory Manager” on page 3 for an overview of the Printer Inventory Manager and how this component fits into your system.

You must customize and start the Printer Inventory Manager before you can create entries in the Printer Inventory and before you can use any other Infoprint Server component. This chapter describes the following required and optional tasks to customize the Printer Inventory Manager:

Task	Condition	See Page:
Enabling the Infoprint Server Feature	Optional: If the Infoprint Server feature was not in your original OS/390 order.	17
Making Run-time Libraries Available	Optional: If libraries are not in LNKLIST	18
Creating HFS Directories	Required	18
Creating the Infoprint Server Configuration File	Optional: To change default configuration	19
Setting Environment Variables	Required	20
Creating Startup and Shutdown Procedures	Optional: To start Printer Inventory Manager with JCL procedure	22
Establishing Security for the Printer Inventory and Operator Commands	Required	21
Enabling Infoprint Server ISPF Panels	Required	24
Backing up the Printer Inventory	Optional: To schedule backup	25

Note: The Printer Inventory *cannot* be shared by any Infoprint Server components running at the *same* or *different* levels on other systems.

After customizing the Printer Inventory Manager, start the Printer Inventory Manager daemon with the **aopstart** command or AOPSTART procedure. Refer to *OS/390 Infoprint Server Operation and Administration* for information about the **aopstart** command.

Enabling the Infoprint Server Feature

If the Infoprint Server priced feature was not included in your original OS/390 order, you must dynamically enable Infoprint Server. To do this, you must add an entry for Infoprint Server to your active IFAPRDxx member of SYS1.PARMLIB. For information on dynamic enablement, refer to *OS/390 Planning for Installation*.

Making Run-time Libraries Available

Infoprint Server requires that the Language Environment® run-time library (CEE.SCEERUN) and the C++ run-time library (CBC.SCLBDLL) reside in an APF-authorized library and be available to Infoprint Server OS/390 UNIX commands, Infoprint Server ISPF panels, Infoprint Server utilities such as PIDU, the AOPPRINT procedure, and IP PrintWay.

IBM recommends that you add the SCEERUN and SCLBDLL data sets to the system LNKLIST. However, if you cannot add the SCEERUN and SCLBDLL data sets to LNKLIST, then you can take the following steps:

- Add the SCEERUN and SCLBDLL data sets to the STEPLIBLIST facility of the BPXPRMxx PARMLIB member.

- Add the following statement to the **/etc/rc** file:

```
export STEPLIB=hlq.SCEERUN:hlq.SCLBDLL
```

- Remove the following statements from the **/etc/profile** file:

```
if -x "$STEPLIB" - && tty -s;
then
    export STEPLIB=none
```

Replace these statements in the **/etc/profile** file with the following statement:

```
export STEPLIB=hlq.SCEERUN:hlq.SCLBDLL
```

- Add the SCEERUN and SCLBDLL data sets to the TSO logon procedures for administrators who use Infoprint Server ISPF panels; see “Defining Libraries in the TSO Logon Procedure” on page 24 for information.
- Add the SCEERUN and SCLBDLL data sets to any JCL that invokes AOPBATCH, such as the AOPPRINT JCL procedure in SYS1.PROCLIB; see “Customizing the AOPPRINT JCL Procedure” on page 35 for information.
- Add the SCEERUN and SCLBDLL data sets to the IP PrintWay startup procedure; see “Creating an IP PrintWay Startup Procedure” on page 88 for more information.

Creating HFS Directories

The Printer Inventory Manager creates the Printer Inventory files and other files in hierarchical file system (HFS) directories. Other components of Infoprint Server also create files in these HFS directories. To create the required HFS directories, follow these steps:

- If the **/etc/Printsrv** directory does not exist, create it using the OS/390 UNIX shell **mkdir** command.
- If the **/var/Printsrv** directory does not exist, create it using the **mkdir** command. IBM strongly recommends that, for easier maintenance, you mount a separate file system at the **/var** mount point and create the **/var/Printsrv** directory in that file system. If you have not created this directory, the **aopsetup** shell script will create it; see “Establishing Security for the Printer Inventory and Operator Commands” on page 21 for information about **aopsetup**.

/var/Printsrv is the default base directory name. You can choose to change the directory name. If you do so, specify the directory name in the **base-directory** statement in the **aopd.conf** configuration file, as described in “Creating the Infoprint Server Configuration File” on page 19.

Creating the Infoprint Server Configuration File

The **aopd.conf** configuration file lets you customize the Printer Inventory Manager and other components of Infoprint Server. This file is optional; if the configuration file does not exist or if a statement in the configuration file is omitted, default values are used.

This section describes the statements in the **aopd.conf** configuration file that are used by Printer Inventory Manager. See “Appendix A. Infoprint Server Files” on page 131 for a description of all the statements you can specify in **aopd.conf** and syntax rules.

To create the configuration file, copy the sample configuration file, **/usr/lpp/Printsrv/samples/aopd.conf**, to **/etc/Printsrv/aopd.conf** with the OS/390 UNIX **cp** command. (You can also use the BPXBATCH command to run the **cp** command.) This is the default location. You can choose to copy the configuration file into another location; however, if you do, specify the full path name of the configuration file in the **AOPCONF** environment variable.

Note: If you change statements in this file while Infoprint Server is running, you must stop and restart the Printer Inventory Manager to pick up the changes. If you change the **inventory** statement, you must also stop and restart other components and products that use this name, including NetSpool, IP PrintWay, and PSF for OS/390.

Specify these statements in the **aopd.conf** configuration file to customize the Printer Inventory Manager.

start-daemons = { }

Identifies the daemons that the **aopstart** command starts. To start only the Printer Inventory Manager daemon, simply code braces without specifying any daemons. The Printer Inventory Manager daemon always starts when you use the **aopstart** command, regardless of which daemons you specify in this statement. By default, the Printer Inventory Manager starts, as well as the Print Interface LPD.

Default: start-daemons = {lpd}

inventory = *inventory*

The name assigned to the Printer Inventory. Specify 1 – 4 uppercase or lowercase letters and numbers; this name is case sensitive. If you start NetSpool or IP PrintWay, specify this name on the EXEC statement of the startup procedure. Also, specify this name in the PSF for OS/390 startup procedure if you want PSF for OS/390 to use information in the Printer Inventory.

Note: IBM recommends that you do *not* change the name of the Printer Inventory in the **inventory** statement after starting the Printer Inventory Manager; if you do, you must stop and restart all Infoprint Server daemons and also stop and restart any other components and products that specify this Inventory name in the startup procedure, including NetSpool, IP PrintWay, and PSF for OS/390.

Default: inventory = AOP1

base-directory = *path*

The directory path in which the Printer Inventory Manager creates Printer Inventory files. Other components of Infoprint Server also create files in this directory.

Attention: Do *not* change this statement while any Infoprint Server daemons are running.

Default: base-directory = /var/Printsrv

ascii-codepage = *codepage*

The name of an ASCII code page supported by OS/390. For code page names, refer to *OS/390 C/C++ Programming Guide*. The Printer Inventory Manager and Print Interface use this code page in the following ways:

- The Printer Inventory Manager displays this code page in the **Printer code page** field in ISPF panels for new IP PrintWay printer definitions.
- Print Interface uses this code page as the source (document) code page when it translates documents received from remote systems, unless a document code page is specified in the printer definition or by the job submitter.
- Print Interface uses this code page as the target (printer) code page when it translates PostScript documents created with a non-ASCII code page.

Default: ascii-codepage = IS08859-1

ebcdic-codepage = *codepage*

The name of an EBCDIC code page supported by OS/390. For code page names, refer to *OS/390 C/C++ Programming Guide*. The Printer Inventory Manager and Print Interface uses this code page in the following ways:

- The Printer Inventory Manager displays this code page in the **Printer code page** field in ISPF panels for new PSF for OS/390 and General printer definitions.
- Print Interface uses this code page as the source (document) code page, when Print Interface cannot determine the user locale for the **lp** command.

Default: ebcdic-codepage = IBM-1047

Setting Environment Variables

Infoprint Server uses environment variables to define its environment within the OS/390 system. Refer to *OS/390 UNIX System Services User's Guide* for a description of environment variables.

You can set these environment variables in the **/etc/profile** file so that they apply for all users on the OS/390 system. If you change the values while any Infoprint Server daemons are running, the settings do not change for the current login shell, daemons, or other processes already started.

These environment variables affect the behavior of Printer Inventory Manager:

AOPCONF The full path name of the Infoprint Server configuration file. This environment variable is *optional*; if you created the configuration file in **/etc/Printsrv/aopd.conf**, you do not need to set this environment variable.

AOPPATH The directory that contains attribute files used by the **lp** command.

This environment variable is *optional*; if you have not created attribute files for the **lp** command, you do not need to set this environment variable. Refer to *OS/390 Infoprint Server User's Guide* for information about attribute files.

LIBPATH	The path used to find dynamic link library (DLL) files. This environment variable is <i>required</i> . If you installed Infoprint Server files in the default directory, add /usr/lpp/Printsrv/lib to any existing values. If you installed Infoprint Server files in a different directory, add the directory to any existing values.						
MANPATH	<p>The path of directories that contain the man pages. This environment variable is <i>required</i>. If the LANG environment variable identifies the language in which you want to view Infoprint Server man pages, add /usr/lpp/Printsrv/man/%L to the values in this variable; otherwise, add one of the following values to any existing values:</p> <table><tr><td>English</td><td>/usr/lpp/Printsrv/man/En_US</td></tr><tr><td>Japanese</td><td>/usr/lpp/Printsrv/man/Ja_JP</td></tr><tr><td>Spanish</td><td>/usr/lpp/Printsrv/man/Es_ES</td></tr></table> <p>Note: Add the new directory path <i>before</i> /usr/man/%L in the MANPATH environment variable, so that the Infoprint Server versions of the lp, lpstat, and cancel man pages are displayed.</p>	English	/usr/lpp/Printsrv/man/En_US	Japanese	/usr/lpp/Printsrv/man/Ja_JP	Spanish	/usr/lpp/Printsrv/man/Es_ES
English	/usr/lpp/Printsrv/man/En_US						
Japanese	/usr/lpp/Printsrv/man/Ja_JP						
Spanish	/usr/lpp/Printsrv/man/Es_ES						
NLSPATH	<p>The path of directories that contain message catalogs. This environment variable is <i>required</i>. If the LANG environment variable identifies the language in which you want to receive Infoprint Server messages, add /usr/lpp/Printsrv/%L/%N to the values in this variable; otherwise, add one of the following values:</p> <table><tr><td>English</td><td>/usr/lpp/Printsrv/En_US/%N</td></tr><tr><td>Japanese</td><td>/usr/lpp/Printsrv/Ja_JP/%N</td></tr><tr><td>Spanish</td><td>/usr/lpp/Printsrv/Es_ES/%N</td></tr></table> <p>%L represents the value of the LANG environment variable. %N is the catalog filename.</p>	English	/usr/lpp/Printsrv/En_US/%N	Japanese	/usr/lpp/Printsrv/Ja_JP/%N	Spanish	/usr/lpp/Printsrv/Es_ES/%N
English	/usr/lpp/Printsrv/En_US/%N						
Japanese	/usr/lpp/Printsrv/Ja_JP/%N						
Spanish	/usr/lpp/Printsrv/Es_ES/%N						
PATH	The path used to locate executables. This environment variable is <i>required</i> . If you installed Infoprint Server executables in the default directory, add /usr/lpp/Printsrv/bin to the existing values. Be sure to add the directory <i>before</i> /bin in the PATH environment variable to ensure that the Infoprint Server versions of the lp , lpstat , and cancel commands are invoked.						

Establishing Security for the Printer Inventory and Operator Commands

You must use the Resource Access Facility (RACF®) or another program that follows system authorization facility (SAF) protocol to establish security for the Printer Inventory and for the Infoprint Server **aopstart** and **aopstop** commands. For more information about RACF, refer to *OS/390 Security Server (RACF) Security Administrator's Guide*.

To establish security using the Resource Access Control Facility (RACF), do the following:

1. Define or update RACF groups for operators and administrators.

You can use existing RACF groups, or you can define new groups. The RACF groups must have an OMVS segment and a group ID (GID) defined. The example in this section assumes that the names of your groups are AOPOPER and AOPADMIN.

AOPOPER A RACF group for operators who need to start and stop Infoprint Server daemons.

AOPADMIN A RACF group for administrators who maintain the Printer Inventory.

Note: You can use one RACF group for both AOPOPER and AOPADMIN if your installation's security policy does not require a distinction between operators and administrators.

2. Create a RACF profile named AOPADMIN to limit access to printer configuration. To do this, follow these steps:
 - a. Define a RACF profile named AOPADMIN under the FACILITY class. You must name this profile AOPADMIN.
 - b. Give the group AOPADMIN read access to the AOPADMIN profile.
3. Run the **aopsetup** shell script to define permissions, specifying the names of the operator group and administrator group you defined in the previous steps. You can run **aopsetup** from an rlogin shell, from an OMVS session, or with the BPXBATCH command. To run the **aopsetup** shell script, you must be a superuser (UID of 0).

For example, assuming that you used the same group names as in the previous steps, enter:

```
/usr/lpp/Printsrv/bin/aopsetup AOPOPER AOPADMIN
```

Creating Startup and Shutdown Procedures

The **aopstart** command starts the Printer Inventory Manager daemon, **aopd**. It also starts any other Infoprint Server daemons specified in the **start-daemons** statement in the **aopd.conf** configuration file.

The **aopstop** command stops the Printer Inventory daemon and any other active Infoprint Server daemons. The **aopstop** command also lets you stop selected Infoprint Server daemons.

The **aopstart** and **aopstop** commands can be entered from the OS/390 UNIX shell, or you can add the **aopstart** command to the **/etc/rc** shell script to start the Printer Inventory Manager (and other daemons) automatically. As an alternative, you can create JCL procedures to invoke the **aopstart** and **aopstop** commands.

This section describes how to create a startup procedure and a shutdown procedure. Refer to *OS/390 Infoprint Server Operation and Administration* for detailed information about the **aopstart** command, Infoprint Server daemons, and how to add **aopstart** to the **/etc/rc** shell script.

AOPSTART Startup Procedure

Infoprint Server provides a startup procedure in SYS1.PROCLIB (AOPSTART), shown in Figure 8 on page 23. This procedure invokes the **aopstart** command to

start the Printer Inventory Manager daemon (**aopd**) and any other daemons specified in the **start-daemons** statement in the **aopd.conf** configuration file.

Notes:

1. The userid assigned to this procedure must be authorized to the AOPOPER RACF profile; see “Establishing Security for the Printer Inventory and Operator Commands” on page 21 for more information.
2. This procedure assumes that environment variables are defined in the **/etc/rc** file.
3. Start the OS/390 SNMP agent *before* starting the SNMP subagent. Refer to *OS/390 SecureWay Communications Server: IP Configuration* for information about how to start OSNMPD.
4. Start TCP/IP *before* starting the Print Interface LPD, the IPP server, or the OS/390 SNMP agent.

```
//*-----  
//*  
//* AOPSTART - This procedure starts the OS/390 Infoprint Server  
//*           daemons. The USERID assigned to this proc must  
//*           be a member of the AOPOPER group.  
//*  
//*-----  
//AOPSTART PROC  
//AOPSTART EXEC PGM=IKJEFT01,PARM='osshell aopstart'  
//SYSPROC DD DSN=SYS1.SBPXEXEC,DISP=SHR  
//SYSTSIN DD DUMMY  
//SYSTSPRT DD SYSOUT=*
```

Figure 8. AOPSTART Startup Procedure — SYS1.PROCLIB(AOPSTART)

AOPSTOP Shutdown Procedure

Infoprint Server provides a shutdown procedure in SYS1.PROCLIB (AOPSTOP), shown in Figure 9 on page 24. This procedure invokes the **aopstop** command to stop all active Infoprint Server daemons after current activity stops.

You can modify this procedure to stop daemons immediately or to stop selected daemons. To do this, specify **aopstop** options in the PARM parameter of the EXEC statement. For example, to stop the Transform Manager and line printer daemons immediately, specify:

```
//... EXEC ... PARM='osshell ostart -d xfd -d lpd now'
```

Refer to *OS/390 Infoprint Server Operation and Administration* for the **aopstop** command options.

Notes:

1. The userid assigned to this procedure must be authorized to the AOPOPER RACF profile; see “Establishing Security for the Printer Inventory and Operator Commands” on page 21 for more information.
2. This procedure assumes that environment variables are defined in the **/etc/rc** file.

```

//*-----
//*
//* AOPSTOP - This procedure stops the OS/390 Infoprint Server
//*          daemons. The USERID assigned to this proc must
//*          be a member of the AOPOPER group.
//*
//*-----
//AOPSTOP PROC
//AOPSTOP EXEC PGM=IKJEFT01,PARM='osshell aopstop'
//SYSPROC DD DSN=SYS1.SBPXEXEC,DISP=SHR
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD DUMMY

```

Figure 9. AOPSTOP Shutdown Procedure — SYS1.PROCLIB(AOPSTOP)

Enabling Infoprint Server ISPF Panels

The administrator can use Infoprint Server ISPF panels to add, browse, copy, edit, and delete printer definitions and other objects in the Printer Inventory. This section describes how to enable the Infoprint Server ISPF panels.

Infoprint Server ISPF panels are available in English, Japanese, and Spanish. The Japanese version of the panels have been translated to code page IBM-939; the Spanish version of the panels have been translated to code page IBM-284.

Defining Libraries in the TSO Logon Procedure

If you have followed the instructions in the ISPF Setup sections in the *OS/390 Program Directory*, you can skip the instructions in this section; otherwise, concatenate the following libraries in the ISPF logon procedure:

- Concatenate one of the following panel libraries to DD statement ISPLLIB, depending on the language in which you want to view the panels.

English AOP.SAOPPENU

Japanese AOP.SAOPPIPN

Spanish AOP.SAOPPESP

- Concatenate one of the following message libraries to DD statement ISPLMLIB, depending on the language in which you want to receive messages.

English AOP.SAOPMENU

Japanese AOP.SAOPMIPN

Spanish AOP.SAOPMESP

- Concatenate library AOP.SAOPEXEC to DD statement SYSPROC or SYSEXEC.

Note: If you have not added the Language Environment run-time library (CEE.SCEERUN) or the C++ run-time library (CBC.SCLBDLL) to the system LNKLIST, concatenate these data sets to the ISPLLIB DD statement, or concatenate them to STEPLIB; the TSOLIB function of TSO/E could be used.

Displaying the Infoprint Server Panel Option

ISPF provides sample ISPF panels to enable panels for most OS/390 elements. The system programmer panel, ISR@390S, includes an entry for Infoprint Server. If you are not using the default ISPF panels, customize your ISPF environment to display the **Infoprint Server for OS/390** option. Refer to the *OS/390 Program Directory* for information about how to use these sample panels to customize your ISPF environment.

Modifying the AOPINIT EXEC

Before using the ISPF panels, review the AOPINIT EXEC provided in AOP.SAOPEXEC(AOPINIT) and modify it, if necessary, for your installation. The AOPINIT EXEC sets default values for fields that display on the Infoprint Server Configuration panel. You can change any of the following values in the AOPINIT EXEC; ISPF users can also change the values for their ISPF sessions on the Configuration panel.

- Configuration file

The path name of the Infoprint Server configuration file. The default file name is **/etc/Printsrv/aopd.conf**.

- Default printer

The name of the printer definition that the **lpstat** command uses as the default printer. Also, the name of the printer definition that **lp** command uses if the **PRINTER** and **LPDEST** environment variables are not set. The default is **lp1**.

- Language

The value of the **LANG** environment variable. The default is **En_US**.

- NLS path

The directory path that contains the Infoprint Server message catalog.

If the **LANG** environment variable identifies the language in which you want to receive Infoprint Server messages, specify **/usr/lpp/Printsrv/%L/%N**; otherwise, add one of the following values:

English **/usr/lpp/Printsrv/En_US/%N**

Japanese **/usr/lpp/Printsrv/Ja_JP/%N**

Spanish **/usr/lpp/Printsrv/Es_ES/%N**

The default is **/usr/lpp/Printsrv/%L/%N**.

- IP PrintWay ISPF trace data set

The name of the data set to contains the trace of IP PrintWay ISPF processing. The default is **userid.ANFISPF.TRACE**.

- IP PrintWay message data set

The name of the data set where IP PrintWay writes messages. The default is **ANF.MESSAGE**.

- Printer Inventory Manager ISPF trace file

The name of the file to contain the trace of ISPF processing. The default file name is **/var/Printsrv/trace**.

To modify AOPINIT, copy AOPINIT to a local library and follow the instructions in the comments in AOPINIT to make your changes. Then add the local library that contains AOPINIT to the **SYSPROC** or **SYSEXEC DD** statement of each user's TSO logon procedure at the beginning of the search order.

Backing up the Printer Inventory

IBM recommends that you back up the Printer Inventory on a regular basis. You can use the Infoprint Server **pidu -c** command to back up the Printer Inventory to a file. To use the **pidu** command, you must be authorized to the **AOPADMIN** RACF profile; see “Establishing Security for the Printer Inventory and Operator Commands” on page 21 for information about the AOPADMIN profile.

For example, to back up the Printer Inventory at 3 AM each day to file `/var/Printsrv/inventory.backup`, use the **crontab** command with the **-e** option to insert the following line into the **crontab** entry for a user authorized to the AOPADMIN profile:

```
00 03 * * * /usr/lpp/Printsrv/bin/pidu -c "export /var/Printsrv/inventory.backup;"
```

For more information about the **crontab** command and the **cron** daemon, refer to *OS/390 UNIX System Services Command Reference*.

Chapter 4. Customizing Print Interface

The Print Interface component of Infoprint Server processes printing requests from local users and remote clients and allocates output data sets on the JES spool. Print Interface supports printing from:

- Local users using OS/390 UNIX shell commands (**lp**, **lpstat**, or **cancel**) or the AOPPRINT JCL procedure
- Remote clients that use TCP/IP LPR to LPD protocol, Internet Printing Protocol (IPP), or Server Message Block (SMB) protocol

See “Print Interface” on page 4 for an overview of Print Interface and how this component fits into your system.

This chapter describes the required and optional customization tasks for Print Interface.

Task	Condition	See Page:
General Customization for Print Interface	Required	27
Customizing for UNIX Printing Commands	Optional: To print with lp command	34
Customizing the AOPPRINT JCL Procedure	Optional: To use AOPPRINT procedure	35
Customizing the Print Interface LPD	Optional: To print from an LPR or with OS/390 Printer Port Monitor for Windows	35
Customizing Print Interface Support for the OS/390 SMB Server	Optional: To print from a Windows client using SMB protocol	37
Customizing the Print Interface IPP Server	Optional: To print from an IPP client	38

After customizing Print Interface, the following tasks must be performed; refer to *OS/390 Infoprint Server Operation and Administration* for information about these tasks:

- Create a printer definition for each target printer in the Printer Inventory and specify printer attributes required by Print Interface.
- Start the Print Interface LPD or IPP server with the **aopstart** command or AOPSTART procedure if you have customized the LPD or IPP server.

General Customization for Print Interface

This section describes general customization tasks you must perform before you submit print requests to Print Interface. In addition, you might need to perform other customization tasks to use specific functions of Print Interface.

Task	Condition	See Page:
Customizing TCP/IP	Required	28
Customizing JES	Required	29
Editing the aopd.conf Configuration File	Optional: To change default configuration	29

Task	Condition	See Page:
Writing a Filter Program	Optional: To inspect or modify input data stream.	30

Customizing TCP/IP

You must customize and start TCP/IP if you plan to submit print jobs to Print Interface from remote systems. You must also customize and start TCP/IP if you plan to use IP PrintWay to transmit print jobs to remote printers. For information about customizing TCP/IP, refer to *OS/390 SecureWay Communications Server: IP Configuration*.

If you plan to use the Print Interface LPD and the OS/390 LPD at port 515 on the same OS/390 system, you must configure multiple TCP/IP stacks. The Print Interface LPD must run on the OS/390 UNIX System Services stack; refer to *TCP/IP for MVS: Planning and Migration Guide* for more information. You can configure the Print Interface LPD to use a port other than 515 if the client programs (LPRs) can send requests to a different port.

You can customize the PROFILE.TCPIP data set when you customize TCP/IP. The PROFILE.TCPIP data set contains statements that initialize the TCP/IP address space. The TCPCONFIG statement in the PROFILE.TCPIP data set updates the TCP layer of TCP/IP. Following is a sample TCPCONFIG statement that you might want to include in the PROFILE.TCPIP data set, followed by an explanation of the parameters and considerations for Print Interface and IP PrintWay.

TCPCONFIG INTERVAL 10 SENDGARBAGE TRUE UNRESTRICTLOWPORTS TCPCVBFRSIZE 32768 TCPSENDBFRSIZE 32768

INTERVAL *minutes* **SENDGARBAGE TRUE**

Number of minutes TCP waits after receiving a packet for a connection before it sends a keep-alive packet for that connection.

IP PrintWay relies on TCP to detect when a connection with the receiving system is no longer usable by sending keep-alive packets; therefore, if you plan to use IP PrintWay, consider specifying a shorter interval than the default, which is approximately two hours. For example, consider specifying an interval of 10 minutes. Note that the interval you specify applies to all TCP applications that direct TCP to send keep-alive packets.

Specify SENDGARBAGE TRUE if any target host requires that the keep-alive packet contain data.

UNRESTRICTLOWPORTS

Ports 1 through 1023 are not reserved.

Print Interface and IP PrintWay must be able to obtain a port in the 0 through 1023 range without having that port explicitly reserved for its use; therefore, specify this parameter to ensure the low ports are not reserved.

TCPCVBFRSIZE *buffersize*

The TCP receive buffer size. IP PrintWay requires a buffer size of at least 32768.

TCPSENDBFRSIZE *buffersize*

The TCP send buffer size. IP PrintWay requires a buffer size of at least 32768.

You can also specify some of these parameters on other TCP/IP statements; the settings from the last statement processed are used.

Note: Also ensure that the ports used by the Print Interface LPD (default port is 515), the IPP server (default port is 631), and the OS/390 SNMP agent (default ports are 161 and 162) are not reserved. For example, if the Print Interface LPD uses port 515, remove or comment out this statement in your PROFILE.TCPIP data set:

```
515 TCP LPSERVE          ; LP Server
```

If you change any of the values in the PROFILE.TCPIP data set, restart TCP/IP to pick up the changes.

See “Customizing TCP/IP” on page 77 for information about how to configure the TCPIP.DATA data set for IP PrintWay.

Customizing JES

The BLNKTRNC parameter on the JES2 OUTCLASS and the TRUNC parameter on the JES3 SYSOUT initialization statement controls whether or not JES truncates trailing blanks in data sets allocated in that output class. The JES2 default is BLNKTRNC=YES; the JES3 default is set in the TRUNC parameter of the JES3 BUFFER statement or TRUNC=YES.

IBM recommends that you specify BLNKTRNC=NO (in JES2) and TRUNC=NO (in JES3) for all output classes in which Print Interface can allocate output data sets, because binary data streams (such as PCL and PostScript) can be corrupted if blanks are removed from the ends of the records written by Print Interface. For example, if Print Interface can allocate jobs on the JES spool in output classes A and C, specify BLNKTRNC=NO (in JES2) and TRUNC=NO (in JES3) for classes A and C.

For information about the JES2 OUTCLASS, refer to *OS/390 JES2 Initialization and Tuning Reference*; for information about the JES2 SYSOUT statement, refer to *OS/390 JES3 Initialization and Tuning Reference*.

Editing the aopd.conf Configuration File

Print Interface uses the following statements in the **aopd.conf** configuration file. Add or edit these statements to customize Print Interface for your installation. See “Appendix A. Infoprint Server Files” on page 131 for more information about the configuration file.

job-prefix = prefix

A prefix added to the job identifier that Print Interface generates for data sets it allocates on the JES spool. This prefix can help the operator identify data sets on the JES spool allocated by Print Interface. Specify two letters, numbers, or national (@ \$ #) characters; the first character *cannot* be numeric.

The Print Interface job identifier is not the same as the JES job identifier. The JES job identifier is assigned by JES and is part of the fully-qualified data set name. The JES job identifier does *not* contain this prefix.

Default: job-prefix = PS

Print Interface also uses the **inventory**, **base-directory**, **ascii-codepage**, and **ebcdic-codepage** statements in the **aopd.conf** configuration file. The Printer

Inventory Manager also uses these statements; see “Creating the Infoprint Server Configuration File” on page 19 for a description.

Writing a Filter Program

Note

This section contains programming-interface information.

Print Interface provides support that lets your installation write filter programs to monitor and modify data before Print Interface writes the data to an output data set on the JES spool. A filter can perform the following types of functions:

- Convert data in a document from one data format to another
- Add a header or trailer page to each document
- Add printer instructions to the beginning of each document
- Add, delete, or modify data in a document
- Terminate processing of a document

You can write two types of filters:

- DLL filter: A filter that resides in a dynamic link library (DLL) and is defined by **cfilter.h**.
- UNIX filter: A filter that reads from **stdin** and writes to **stdout**.

Notes:

1. To add header pages and trailer pages to output data sets that IP PrintWay transmits to the target printer, instead of writing a filter, you can write an IP PrintWay Begin Data Set and an IP PrintWay End Data Set exit routine; see “Begin Data Set Exit” on page 102 and “End Data Set Exit” on page 111 for more information. Also, the administrator can specify printer instructions for IP PrintWay to add to the beginning and end of each data set in a printer definition. Refer to *OS/390 Infoprint Server Operation and Administration* for more information.
2. To add header or trailer pages to output data sets that PSF for OS/390 processes, you can write a PSF for OS/390 separator page exit. Refer to *PSF for OS/390: Customization* for information.

Filter Options

You can pass filter options and operands to your DLL or UNIX filter. The administrator can specify filter options and operands in the printer definition, and the user can specify filter options in the **filter-options** job attribute.

The administrator can use the special % operator to pass the value of a job attribute to the filter. For example, if a user specifies a job attribute on the **lp** command, the value of the attribute can be passed to the filter. Use the following format:

`%attribute_name`

where:

- % Causes the entire `%attribute_name` substring to be replaced by the value of the attribute.

attribute-name

Is the name of any job attribute, as defined in *OS/390 Infoprint Server User's Guide*.

If an attribute value is empty or contains special character or spaces, the value is enclosed in quotation marks and passed to the filter as one argument.

Examples: Example 1: Using the % operator

In this example, **%address-text** is specified as an operand for UNIX filter **test_filter** in the printer definition:

```
spawn /usr/local/bin/test_filter %address-text
```

If the **lp** command specifies the following **address-text** job attribute:

```
lp -o "address-text={a1 a2}"
```

Filter **test_filter** is passed these arguments:

```
/usr/local/bin/test_filter "{ a1 a2 }"
```

The filter program receives `/usr/local/bin/test_filter` as the first argument and `{ a1 a2 }` as the second argument (without quotes).

Example 2: Using the % operator with a filter written in the REXX language

Sometimes the values for job attributes are null. If you use REXX to write your filter, use a **parse** statement like the following one to parse arguments, so that null arguments are properly passed to the program:

```
parse arg arg1 ' ' arg2 ' ' arg3
```

Sometimes the values for job attributes contain embedded spaces. Because the **parse** statement shown above does not correctly receive an argument that contains embedded spaces, use an alternative delimiter such as `/`, to delimit filter options that use the **%** operator; ensure that the delimiter you select does not have a special meaning to the shell and does not occur in the data. For example, if the administrator uses `/` to delimit filter options in the printer definition as shown:

```
spawn /usr/local/bin/test_filter /%name-text/%document-format/
```

And, if the **lp** command specifies the **name-text** job attribute:

```
lp -o "name-text='mike smith'" text.file
```

The command, after substitution of the attribute values, is:

```
/usr/local/bin/test_filter /"mike smith"/text/
```

You can use the following REXX **parse** statement to parse the arguments:

```
parse arg '/' name_text '/' doc_format '/'
```

For more information about how to specify the filter name in a printer definition, see “Invoking a DLL or UNIX Filter” on page 33 and refer to *OS/390 Infoprint Server Operation and Administration*.

Writing a DLL Filter

To assist you in writing a new DLL filter, IBM provides the source code for the **aopfilter.so** DLL filter in the following files:

- **cfilter.h**, a header file that contains declarations and interface descriptions. Do not change the contents of this file.

- **cfilter.c**, the source code for the **aopfiltr.so** filter.

Both files are in directory **/usr/lpp/Printsrv/samples**. For more information about **aopfiltr.so**, see “Filters Provided With Infoprint Server and Infoprint Server Transforms for OS/390” on page 34.

To customize the **aopfiltr.so** filter, do the following:

- Modify file **cfilter.c** as necessary, using the comments in the prologs of the file.
- Link the filter as a dynamic link library (DLL), as described in the comments in file **cfilter.c**.

Input to a DLL Filter: A DLL filter has access to the following information:

- Filter options and operands.
- Environment variables.
- Input data in a document; data in text format has already been converted to the printer code page.
- Name of the printer code page used to convert the data.
- Format of the data, as determined by Print Interface or as specified by the user.
- The LPD control file in a format defined by RFC 1179.
- The type of carriage controls in the input file.
- The name of the input file.
- The name of the printer definition for the print request.

Output from a DLL Filter: A DLL filter can return the following output to Print Interface:

- Modified data.
- Format of the data.
- The type of carriage controls.
- Job attributes. For a list of Infoprint Server job attributes, refer to *OS/390 Infoprint Server User's Guide*. Job attributes specified by the job submitter (for example, on the **lp** command), override the same job attributes set by the filter.
- A return code. If the return code indicates an error, Print Interface ends processing of the document.

Writing a UNIX Filter

A UNIX filter can be a shell executable, such as a C program, REXX exec, shell script, and so on. Print Interface spawns a UNIX filter as a separate process. (**cat** found on all UNIX systems is an example of a UNIX filter.)

The shell script in Figure 10 shows an example of a simple UNIX filter:

```
#!/bin/sh
# DESCRIPTION:
#   This filter processes EBCDIC (not ASCII).
#   Terminate each line with "carriage return + line feed".
#=====
awk '{ printf "%s\r\n", $0 }';
exit
```

Figure 10. Sample UNIX Filter

Input to a UNIX Filter: A UNIX filter can access the following information:

- Filter options and operands.

- Environment variables.
- The input data in **stdin**; data in text format has already been converted to the printer code page.

Output from a UNIX Filter: A UNIX filter can return the following output to Print Interface:

- Modified data in **stdout**.
- A return code. If the return code indicates an error, Print Interface ends processing of the document and does not allocate the data set on the JES spool.

Invoking a DLL or UNIX Filter

To request that Print Interface invoke a filter, the administrator must specify the filter name and filter options in the printer definition for the target printer. The full path name is required unless the filter is in a directory named in the LIBPATH environment variable (for DLL filters) or in the PATH environment variable (for UNIX filters). For UNIX filters, the administrator must type **spawn** before the filter name.

The administrator specifies the filter in the **Filter** field of the Processing section of a printer definition, or in a Processing component. The administrator can specify different filters for different input data formats. Refer to *OS/390 Infoprint Server Operation and Administration* for more detailed information.

Programming Considerations

Consider the following points when coding a filter program:

- You should be an experienced programmer who is aware of overall system implications.
- Do not issue a long-term explicit or implied WAIT in the filter program because it can reduce the throughput of the Print Interface LPD or IPP server.
- The filter program runs in 31-bit addressing mode.
- Code the filter program to be reentrant.
- All filter programs run in problem state.
- Programming exceptions cause Print Interface to abend; an ESTAE is in effect while a filter program is running.
- A DLL filter, in the initialization function, can obtain a work area and pass a pointer to that work area to other functions in the filter program. Using the work area, one function can pass data to another function.

Note: Future Print Interface program maintenance may require that you recompile your filter programs.

Testing Filters

To test your filter, specify the name of the filter in a printer definition and use the **lp** command to print a document to that printer definition. As an alternative, you can use the **filter** command to test your filter; refer to **/usr/lpp/Printsrv/samples/cfilter.h** for a description of the **filter** command.

Modifying Filters

After modifying a DLL filter, you might need to restart the Print Interface LPD or the Print Interface IPP server to cause them to load the new version of the filter. This is necessary only if the LPD or IPP server has already used the old version of the filter with the same name.

To restart the LPD and IPP server, use the **aopstop -d lpd -d ippd** command to stop the LPD and IPP server and then use the **aopstart** command to start them again. Refer to *OS/390 Infoprint Server Operation and Administration* for more information about these commands.

You do *not* need to restart the LPD and IPP server when you modify a UNIX filter or if you have written a new filter and tested it only with the **lp** command or the **filter** command.

Filters Provided With Infoprint Server and Infoprint Server Transforms for OS/390

Infoprint Server provides two DLL filters in directory **/usr/lpp/Printsrv/lib**:

- **aopfiltr.so**, which converts ASCII line-feed controls (X'0A') that are not preceded by carriage-return controls to carriage-return and line-feed controls (X'0D0A'). The X'0D0A' control is suitable for most ASCII printers and print queues. IBM recommends that the administrator specify the **aopfiltr.so** filter in the printer definition for ASCII printers for text data.
- **aoprform.dll**, which transforms PCL, PostScript, and PDF data remotely using Infoprint Manager for AIX or PSF for AIX.

Infoprint Server Transforms for OS/390 provides three DLL filters in directory **/usr/lpp/Printsrv/lib**:

- **pcl2afp.dll**, which transforms PCL data to AFP format
- **pdf2afp.dll**, which transforms PostScript and PDF data to AFP format
- **sap2afp.dll**, which transforms SAP data to AFP format

Customizing for UNIX Printing Commands

Print Interface provides enhanced OS/390 UNIX shell commands for printing: **lp**, **lpstat**, and **cancel**. To use these commands, no additional customization is required and you do not need to start any daemons in addition to the Printer Inventory Manager daemon. However, you can optionally set the following environment variables in the **/etc/profile** file:

AOOPTIONS

Specifies a string of job attributes and values that are to be in effect for each **lp** command. The **lp** command includes the attributes specified in this environment variable before the attributes specified, if any, on the **-o** option of the **lp** command. Because the **lp** command reads the value of the AOOPTIONS environment variable before the options you specify on the command line, a user can override the values of this variable. This variable is *optional*; use this environment variable to specify job attributes that are constant for all print jobs. Refer to *OS/390 Infoprint Server User's Guide* for the format of the job attributes and values.

PRINTER or LPDEST

The default printer for the **lp** command. The value in LPDEST overrides the value in PRINTER.

This variable is *optional*. You can also use Infoprint Serve ISPF panels to define a default printer; see *OS/390 Infoprint Server Operation and Administration*. The printer named in either LPDEST or PRINTER overrides the printer named on the ISPF configuration panel.

Customizing the AOPPRINT JCL Procedure

Print Interface provides the AOPPRINT JCL procedure in SYS1.PROCLIB so that local users can submit the **lp** command from a batch job. Refer to *OS/390 Infoprint Server User's Guide* for information about how to use the AOPPRINT JCL procedure.

Figure 11 shows the AOPPRINT procedure.

```
/*-----  
/* AOPPRINT - OS/390 Print Server batch print procedure  
/*-----  
/*AOPPRINT PROC PRINTER='p1',OPTIONS=,OUTCLASS='*',ERRCLASS='*'  
//LP      EXEC PGM=AOPBATCH,  
// PARM='/lp -d &PRINTER -o "&OPTIONS" //DD:SYSIN'  
//STDOUT  DD  SYSOUT=&OUTCLASS  
//STDERR  DD  SYSOUT=&ERRCLASS  
/* STDENV may point to a dataset containing environment variables.  
/*      Builtin values will work for the default installation.  
/*STDENV  DD  DSN=USERID.JCL(ENVVARS),DISP=SHR
```

Figure 11. AOPPRINT JCL Procedure in SYS1.PROCLIB

You can customize the AOPPRINT JCL procedure for your installation in the following ways:

- To change the name of the default printer definition, specify the printer definition name in the PRINTER option on the PROC statement. If you don't change the name, the default is **p1**. The administrator should create a printer definition with the default name.
- To define environment variables used by the Printer Inventory Manager, define environment variables in an STDENV DD statement, as shown in the procedure. If your installation installed Infoprint Server files in the default directory, **/usr/lpp/Printsrv/**, then you do not need to specify the LIBPATH, NLSPATH, and PATH environment variables. To define environment variables, specify each variable on a separate line, in the format *variable=value*, as follows:
LIBPATH=/usr/mylib
PATH=/usr/mylib
- If you have not added the Language Environment run-time library (CEE.SCEERUN) or the C++ run-time library (CBC.SCLBDLL) to the system LNKLIST, concatenate these data sets to a STEPLIB DD statement.

Customizing the Print Interface LPD

The Print Interface Line Printer Daemon (LPD) accepts print requests from remote clients that use the LPR to LPD protocol as defined by RFC 1179. The **lpr** and **lpq** commands (supported on various platforms such as MVS, VM, AS/400®, OS/2®, AIX, and UNIX), the AIX **enq** and **qstat** commands, and the OS/390 Printer Port Monitor for Windows use the LPR to LPD protocol. The Print Interface LPD allocates jobs on the JES spool and can provide notification and job status.

This section describes the following required and optional tasks to customize the LPD:

Task	Condition	See Page:
Editing the aopd.conf Configuration File	Optional: To change default configuration	36

Task	Condition	See Page:
Customizing the LPR	Required	36

After customizing the LPD, you must start the LPD using the **aopstart** command or AOPSTART procedure. Refer to *OS/390 Infoprint Server Operation and Administration* for information about the **aopstart** command.

Editing the aopd.conf Configuration File

Add or edit these statements in the **aopd.conf** configuration file. See “Appendix A. Infoprint Server Files” on page 131 for more information about the configuration file.

start-daemons = { lpd }

To start the LPD, add **lpd** to the values in this statement before issuing the **aopstart** command. Enclose the values in braces. By default, the Printer Inventory Manager and the LPD start.

Default: start-daemons = {lpd}

lpd-port-number = nnnn

The number of the port at which the Infoprint Server LPD waits for print requests. Port 515 is the well-known port for communication between LPRs and LPDs. If you specify a port other than 515, ensure that the port is not used by any other service on the OS/390 system and that you customize the LPRs to send print requests to the new port. Windows users must specify this port when configuring the OS/390 printer port monitor on a Windows system. This statement is optional.

Note: This port number must not be reserved in the PROFILE.TCPIP data set. See “Customizing TCP/IP” on page 28 for more information.

Default: lpd-port-number = 515

Customizing the LPR

To submit a print request from an LPR to the Print Interface LPD, a user typically uses a TCP/IP command, such as the UNIX **lpr** command. A Windows user can also use the OS/390 Printer Port Monitor for Windows.

To submit a print request or to configure the OS/390 Printer Port Monitor for Windows, the user might need the following information:

- IP address or host name of the OS/390 system where the Print Interface LPD is running.
- Port number at which the Print Interface LPD is listening.
- Name of the printer definition for the target printer defined in the Printer Inventory.
- The printer driver to use for the target printer. For printing to IBM AFP printers, Windows users can download the AFP Printer Driver for Windows.

On some platforms, the user can specify whether or not the LPR must use a port in the range of 721 to 731. The Print Interface LPD does not require that the LPR use a port in this range.

The Print Interface LPD supports these command codes that can be specified in the LPD control file:

- H; this value becomes the address in the Infoprint Server **address-text** job attribute.
- J; this value becomes the title in the Infoprint Server **title-text** job attribute, unless T is also specified.
- M; this value activates notification.
- N; this value becomes the filename displayed by the **lpq** or **lpstat** command.
- P; this value becomes the job owner. Print Interface allocates data sets on the JES spool with the job owner in the job name field (see note).
- T; this value becomes the title in the Infoprint Server **title-text** attribute.
- X; Infoprint Server job attributes. If the attributes are prefixed with AOP, the LPD validates that the job attributes are supported by the target printer.
- f, l, p, c, d, g, n, r, t, and v.

The Print Interface LPD ignores all other command codes. See RFC 1179 for a description of these LPD command codes.

Note: JES does not permit the Print Interface LPD to allocate data sets on the JES spool with the owner name that is specified in the P control code; therefore, to make the owner name available in SMF records and in the data set name displayed by SDSF, the Print Interface LPD uses the owner name as the job name when allocating data sets on the JES spool.

For additional information about how to use submit jobs from remote systems, how to configure the OS/390 Printer Port Monitor for Windows, and Infoprint Server job attributes, refer to *OS/390 Infoprint Server User's Guide*.

Customizing Print Interface Support for the OS/390 SMB Server

Print Interface, in conjunction with support provided by the OS/390 SMB server, lets users print from Windows workstations that use Server Message Block (SMB) printing protocol. Jobs submitted from the client can be printed on any printer defined in the Printer Inventory. Print Interface allocates output data sets on the JES spool and provides job status.

As an alternative to using SMB protocol, users can install the OS/390 Printer Port Monitor for Windows. See “Chapter 9. Installing and Customizing the Windows Client” on page 127 for information that can help you decide whether to use SMB protocol or the Port Monitor.

To use the SMB support in Print Interface, you must apply Infoprint Server PTF UW64005 (for APAR OW41376). No additional customization steps are required in Print Interface. You must start the Printer Inventory Manager daemon; it is not necessary to start any other Infoprint Server daemons.

You must, however, customize the OS/390 SMB server as described in *OS/390 Distributed File Service SMB Administration Guide and Reference*. For each printer definition defined in the Infoprint Server Printer Inventory, you must define a shared printer to the OS/390 SMB server. When you create a shared printer, consider the following:

- You must specify a name for each shared printer. You can use either the same name as the name of the printer definition or a different name. Using the same name can make it easier to locate the printer in the Printer Inventory.

- You can also specify the printer type when you create a shared printer. The printer type designates the printer driver for the printer, which must reside on the client's system. You can download IBM printer drivers from the Web site: **www.ibm.com/printers**.

If the target printer is an AFP printer, you can specify the AFP printer driver for Windows. As an alternative, if the administrator has configured printer definitions in the Printer Inventory to use the PCL or PostScript (and PDF) to AFP data transforms, then you can specify a generic PCL or PostScript driver for AFP printers. Refer to *OS/390 Infoprint Server Operation and Administration* for information about how to configure printer definitions so that Print Interface uses data transforms.

Customizing the Print Interface IPP Server

Print Interface provides an Internet Printing Protocol (IPP) server, which can receive print requests from IPP clients that use the IPP. The Print Interface IPP server allocates jobs on the JES spool and provides job status. The IPP server does not support canceling a print request.

Note: The IP PrintWay IPP client can transmit output data sets from the JES spool to IPP servers running in a remote printer or host system. To use the IP PrintWay IPP client, you do not need to configure the Print Interface IPP server. To use the IP PrintWay IPP client, however, the administrator must define attributes for IPP printing in the printer definitions in the Printer Inventory.

This section describes the following required and optional tasks to customize the IPP server:

Task	Condition	See Page:
Editing the aopd.conf Configuration File	Required	38
Setting Environment Variables	Optional: To change default configuration	39
APF-Authorizing Libraries	Required	19
Customizing the IPP Workstation Client	Required	39

After customizing the IPP server, you must start the IPP server. Refer to *OS/390 Infoprint Server Operation and Administration* for information about the **aopstart** command.

Editing the aopd.conf Configuration File

Add or edit the following statements in the **aopd.conf** configuration file. See “Appendix A. Infoprint Server Files” on page 131 for more information about the configuration file.

ipp-port-number = nnnn

The number of the port at which the IPP server waits for print requests. Port 631 is the well-known port for communication between IPP clients and IPP servers. If you specify a port other than 631, ensure that the port is not used by any other service on the OS/390 system. This statement is optional.

Note: This port number must not be reserved in the PROFILE.TCPIP data set. See “Customizing TCP/IP” on page 28 for more information.

Default: ipp-port-number = 631

start-daemons = { ippd }

To start the IPP server daemon, add **ippd** to any other values in this statement before issuing the **aopstart** command. Enclose the values in braces. By default, only the Printer Inventory Manager and the LPD start. This statement is required to start the IPP server.

Default: start-daemons = {lpd}

Setting Environment Variables

The following environment variable affects the behavior of the IPP server. You can set this environment variable in the **/etc/profile** file.

CLASSPATH

The full path names of Infoprint Server IPP files. If you installed Infoprint Server IPP files in the default directory, **/usr/lpp/Printsrv**, you do not need to modify this environment variable. If you installed Infoprint Server files in a different directory, add the following values to any existing values in this variable:

- **directory/classes/ipp.jar**
- **directory/classes/ippserver.jar**
- **directory/classes/ippclient.jar**
- **directory/classes/ippreal.jar**

where *directory* is the directory that contains the files.

JAVA_HOME

The path used to locate Java files. This environment variable is *optional*; if you installed Java files in the default directory, **/usr/lpp/java/J1.1**, you do not need to set this environment variable.

APF-Authorizing Libraries

When using the IPP server, the Java libraries **libjava.a**, **libjtc.so**, and **libzip.so** must be marked as APF-authorized (that is, residing in an authorized library).

If the **JAVA_HOME** environment variable names the Java directory, you can use the following OS/390 UNIX shell commands to APF-authorize these libraries:

```
cd $JAVA_HOME/lib/mvs/native_threads
extattr +a libjava.a libjtc.so libzip.so
```

You must be permitted to the BPX.FILEATTR.APF facility class in order to run the **extattr** command. A sample job is provided in **SYS1.SAMPLIB(AOPJAUTH)** to perform this function.

Note: When maintenance is applied to Java, APF authorization is lost; therefore, you must rerun **AOPJAUTH** after maintenance is applied to Java.

Customizing the IPP Workstation Client

To submit a print request from an IPP client that is running on a workstation, the user must specify the Universal Resource Indicator (URI) of the target printer. For printers defined in a printer definition in the Printer Inventory, use one of the following URIs:

- **ipp://host[:port]/servlet/IPPServlet/printername**
- **http://host[:port]/servlet/IPPServlet/printername**

where:

host The host name or IP address of the OS/390 system.

port The port number where the IPP server is listening. Specify the same number as specified in the **ipp-port-number** statement in the **aopd.conf** file.

Defaults:

- If the **ipp:** format is used, the default port is 631.
- If the **http:** format is used, the default port is 80.

printername The name of the printer definition in the Printer Inventory.

Examples: If the IPP server is listening at port 631, the IPP client can use either one of these URIs:

ipp://9.99.1234/servlet/IPPServlet/myprinter

http://9.99.1234:631/servlet/IPPServlet/myprinter

Note: The Print Interface IPP server cannot receive data that has been encrypted; therefore, the user should ensure that the IPP client does not encrypt data.

An IPP client can specify job attributes with a print request. The Print Interface IPP server supports some of the possible input job IPP attributes. Other input IPP attributes are ignored unless the target printer or print server contains an IPP server. In this case, IP PrintWay transmits all input job IPP attributes to the target IPP server.

Table 3 lists the IPP job attributes that the Print Interface IPP server supports for all printers and the corresponding Infoprint Server job attribute that is set. Refer to *OS/390 Infoprint Server User's Guide* for information about the Infoprint Server job attributes.

Table 3. IPP Attributes and Corresponding Infoprint Server Attributes

IPP Job Attribute	Corresponding Infoprint Server Job Attribute
copies	copies
job-hold-until	hold
job-name	title-text
job-originating-user-name or requesting-user-name ¹	owner ²
job-priority	jes-priority
media	forms
printer-resolution	resolution
sides	duplex

1. **requesting-user-name** is used only if **job-originating-user-name** is not specified.
2. JES does not permit the Print Interface LPD to allocate data sets on the JES spool with the owner name that is specified in the IPP job attribute; therefore, to make the owner name available in SMF records and in the data set name displayed by SDSF, the Print Interface LPD uses the owner name as the job name when allocating data sets on the JES spool.

For additional information about how to submit print requests from an IPP client, refer to *OS/390 Infoprint Server User's Guide*.

Chapter 5. Customizing Infoprint Server Transforms and the Transform Manager

Infoprint Server Transforms, an IBM program product, provides the following transforms for the OS/390 system:

- Printer Control Language (PCL) to AFP
- PostScript and Portable Data Format (PDF) to AFP
- SAP Output Text Format (OTF) versions 1 and 2, and SAP Advanced Business Application Programming (ABAP) versions 1 and 2 to AFP

You must use the Infoprint Server Transforms Program Directory to install Infoprint Server Transforms.

The Transform Manager component of Infoprint Server manages the PCL and PostScript/PDF transform daemons provided by Infoprint Server Transforms. Note that the same transform daemon transforms both PostScript and PDF data to AFP format. See “Infoprint Server Transforms for OS/390 and the Transform Manager” on page 7 for an overview of Infoprint Server Transforms and the Transform Manager and how these components fit into your system.

This chapter describes the following required and optional tasks to customize Infoprint Server Transforms and the Transform Manager:

Task	Condition	See Page:
Enabling Infoprint Server Transforms	Required	44
Creating the aopxfd.conf Configuration File	Required for PCL, PostScript, and PDF to AFP transforms	44
Editing the aopd.conf Configuration File	Required for PCL, PostScript, and PDF to AFP transforms	48
Setting Environment Variables	Optional: To change default location of files	49
Installing Multibyte Conversion Tables for SAP to AFP Transform	Required for SAP to AFP transform of DBCS data	49
Customizing SAP to AFP Configuration Files	Optional: To change default configuration for SAP to AFP transform	50

After you customize the Transform Manager, the following administrative tasks must be performed; refer to *OS/390 Infoprint Server Operation and Administration* for information about these tasks.

- Specify transform filters in printer definitions if you want Print Interface to transform data to AFP before placing it on the JES spool. If your installation plans to use only the OS/390 UNIX commands (**pcl2afp**, **pdf2afp**, **ps2afp**, and **sap2afp**) this task does not apply.
- Start the Transform Manager with the **aopstart** command or AOPSTART procedure. Refer to *OS/390 Infoprint Server Operation and Administration* for information about the **aopstart** command. If your installation plans to use only the SAP to AFP transform, this task does not apply.

Note: Instead of transforming data to AFP on the OS/390 system, Print Interface can transform data to AFP remotely on an AIX system. To use AIX transforms, you do not need to customize Infoprint Server Transforms or the Transform Manager; however, Infoprint Manager for AIX (or PSF for AIX) must be installed on an AIX system, and the administrator must specify the remote transform filter in printer definitions in the Printer Inventory; refer to *OS/390 Infoprint Server Operation and Administration* for more information.

Enabling Infoprint Server Transforms

To use transforms provided with Infoprint Server Transforms, all installations must dynamically enable Infoprint Server Transforms. To do this, add the following entry to your active IFAPRDxx member of SYS1.PARMLIB if it has not already been added:

```
PRODUCT OWNER('IBM CORP')
      NAME('INFOPRINT XFORMS')
      ID(5697-F51)
      VERSION(*) RELEASE(*) MOD(*)
      FEATURENAME('TRANSFORM TO AFP')
      STATE(ENABLED)
```

For more information on how to use OS/390 dynamic enablement, refer to *OS/390 MVS Product Management*.

Creating the aopxfd.conf Configuration File

The transform configuration file contains required configuration information that the Transform Manager uses to manage the PCL to AFP and PostScript/PDF to AFP transforms. You do not need to create this configuration file to use SAP to AFP transforms. You must create this configuration file before starting the Transform Manager.

To create the configuration file, copy the sample configuration file provided in `/usr/lpp/Printsrv/samples/aopxfd.conf` to `/etc/Printsrv/aopxfd.conf`, using the OS/390 UNIX System Services **cp** command. (You can also use the BPXBATCH command to issue the **cp** command.) This is the default location. You can choose to copy the configuration file into another location; if you do so, specify the full path name of the configuration file in the **AOPXFD_CONF** environment variable.

If you edit the configuration file while the Transform Manager is started, use the **aopstop -d xfd** command to stop the Transform Manager and then use the **aopstart** command to restart it, so that your changes take effect. Refer to *OS/390 Infoprint Server Operation and Administration* for more information about how to stop and start the Transform Manager.

PCL to AFP Transform Classes

For PCL to AFP transforms, you can define transform classes in the configuration file. Transform classes are optional, but you must define transform classes if you plan to print on printers with different resolutions and paper sizes. The sample transform configuration file, `/usr/lpp/Printsrv/samples/aopxfd.conf`, defines several transform classes.

To use a transform class, the administrator can specify the class in the printer definition, the user can specify the class on the **pcl2afp** command, **lp** command, and AOPPRINT procedure; see the description of the **transform** statement in “Format” on page 45 for more information.

Format

In the **aopxfd.conf** file, specify a transform entry for each transform you plan to use. Each entry consists of a set of configuration statements, starting with the **transform** statement and ending with a semicolon.

```
# comment
transform transformname[_transformclass]
start-command = "daemon [ option ]..."
[ start-directory = path ]
[ maximum-idle-time = seconds ]
[ minimum-active = number ]
[ maximum-active = number ]
[ environment = { name -> value [ name -> value]... } ] ;
```

transform transformname[_transformclass]

This statement is required and must be the first statement in the entry. It identifies the start of a transform entry.

transformname

The name of the transform. To use transforms provided by Infoprint Server Transforms, specify one of the following names:

- **pcl2afp**, the name of the PCL to AFP transform
- **ps2afp**, the name of the PostScript (and PDF) to AFP transform

transformclass

The name of a transform class. This parameter does not apply for the PostScript (and PDF) to AFP transform; it is optional for the PCL to AFP transform.

Specify from 1 to 63 characters, including letters, numbers, or special characters. To use this transform class, the administrator names the class in the **-c** option of the **pcl2afp.dll** filter in the printer definition in the Printer Inventory. Also, a user can name the class in the **-c** option of the **pcl2afp** command, and in **filter-options** attribute of the **lp** command and AOPPRINT JCL procedure.

Default: If this parameter is omitted, this entry is used only if the filter or command does not name a class.

Example: transform pcl2afp_letter_300

To use this transform class on the **pcl2afp** command, type:

```
pcl2afp -c letter_300
```

Default: None

start-command = "daemon [option]..."

The name of the transform daemon and any options accepted by the transform daemon. This statement is required. To use a transform daemon provided by Infoprint Server Transforms, specify one of the following values:

- **pcl2afpd**, the PCL to AFP transform daemon
- **ps2afpd**, the PostScript (and PDF) to AFP transform daemon

If the transform daemon is not in a directory identified in the PATH environment variable, then specify the full directory path name of the daemon.

The **pcl2afpd** daemon supports the following option:

-m *nnn*{K | M}

The number of bytes of memory the transform daemon can use to perform transforms. Specify the number of bytes in either kilobytes or megabytes. The amount required depends on the compression type, the image being compressed, fonts, and so on.

Use the following algorithm or refer to the following table to determine the amount to initially specify; if the transform fails due to lack of memory, then specify a larger amount of storage.

$((\text{width in pixels} / 8) * \text{height in pixels}) * 3$

Width in pixels = (width in inches * resolution)

Height in pixels = (height in inches * resolution)

The following table shows the amount of storage for different paper sizes and resolutions (240, 300, and 600 pixels per inch), using the previous algorithm:

Paper Size	240	300	600
Letter	2.0M	3.2M	12.6M
Legal	2.6M	4.0M	16.1M
A4	2.1M	3.3M	13.1M

Example: start-command = "pcl2afpd -m 4M"

Default: 6M

Note: If you specify this option, enclose the keyword and value in single or double quotation marks.

Default: None

start-directory = *path*

Path where the fonts and resources used by the PostScript (and PDF) to AFP transform are located. This statement does not apply for the PCL to AFP transform; it is optional for the PostScript (and PDF) to AFP transform. If you installed Infoprint Server Transforms fonts and resources in the default directory, specify **/usr/lpp/Printsrv/ps2afp**.

Default: Current directory

maximum-idle-time = *seconds*

The number of seconds before the Transform Manager shuts down an idle transform daemon and system resources are freed. This statement is optional; however, to avoid having many transform daemons active, consider specifying either this statement or the **maximum-active statement** (or both). The number you specify must be greater than 0.

Default: An idle transform daemon is not shut down.

minimum-active = *number*

The number of transform daemons that the Transform Manager starts, and also the minimum number of transform daemons that the Transform Manager keeps active (that is, not shut down), even when the maximum-idle-time expires for an idle transform daemon. The number you specify must be greater than 0. This statement is optional.

Default: minimum-active = 0

maximum-active = *number*

The maximum number of transform daemons that the Transform Manager keeps active. When this number is reached, the Transform Manager does not start a new transform daemon to perform the transform; therefore, the transform request waits until a transform daemon is available. This statement is optional; however, to avoid having many transform daemons active, consider specifying either **maximum-active** or **maximum-idle-time**, or both. The number you specify must be greater than 0 and greater than or equal to the number specified in **minimum-active**.

Default: No maximum number; transform daemons are started when needed.

environment = {*name* -> *value* [*name* -> *value*...]... }

Environment variables that define the transform environment for PCL to AFP transforms. Enclose the environment variables in braces. The values in these environment variables override values of environment variables with the same names that were set when the **aopstart** command was issued. Following are the environment variables you can specify:

Environment Variable	Meaning
AOP_PAGE_HEIGHT	Page height in inches (i) or millimeters (m). Default: AOP_PAGE_HEIGHT -> 11i
AOP_PAGE_WIDTH	Page width in inches (i) or millimeters (m). Default: AOP_PAGE_WIDTH -> 8.5i
AOP_RESOLUTION	Output resolution in pixels per inch. Default: AOP_RESOLUTION -> 240
AOP_HORIZONTAL_MARGINS	Left and right margins in inches (i) or millimeters (m). Default: AOP_HORIZONTAL_MARGINS -> 0.167i
AOP_VERTICAL_MARGINS	Top and bottom margins in inches (i) or millimeters (m). Default: AOP_VERTICAL_MARGINS -> 0.167i

Example

This example shows three sample entries in the **aopxfd.conf** configuration file:

- The first entry is for the PostScript to AFP transform and also the PDF to AFP transform.
- The second entry is for the PCL to AFP transform and shows a transform option. Because the **transform** statement does not specify a transform class, the

Transform Manager uses this entry when the administrator and user do not specify a transform class as a filter option.

This entry sets environment variables for letter size paper (8.5 inches wide, 11 inches high) and printers with a resolution of 240 pixels per inch.

- The third entry is also for the PCL to AFP transform. The Transform Manager uses this entry when the administrator or user specifies transform class **letter_300** as a filter option.

This entry sets environment variables for letter size paper (8.5 inches wide, 11 inches high) and printers with a resolution of 300 pixels per inch.

When the Transform Manager starts, it starts three transform daemons, one for the PostScript/PDF to AFP transform, one for the PCL to AFP transform, and one for the PCL to AFP letter_300 class transform. The Transform Manager keeps no more than two of each type of transform daemon active at any time. The Transform Manager shuts down any transform daemon that is idle and frees system resources after 5 minutes; however, the Transform Manager keeps at least one transform daemon of each type active.

```
transform ps2afp
  start-command = ps2afpd
  start-directory = /usr/lpp/Printsrv/ps2afp
  minimum-active = 1
  maximum-active = 2
  maximum-idle-time = 300 # 5 minutes
;

transform pcl2afp
  start-command = "pcl2afpd -m 4M"
  minimum-active = 1
  maximum-active = 2
  maximum-idle-time = 300 # 5 minutes
  environment = {
    AOP_PAGE_HEIGHT -> 11i
    AOP_PAGE_WIDTH -> 8.5i
    AOP_RESOLUTION -> 240
    AOP_HORIZONTAL_MARGINS -> 0.167i
    AOP_VERTICAL_MARGINS -> 0.167i
  }
;

transform pcl2afp_letter_300
  start-command = pcl2afpd
  minimum-active = 1
  maximum-active = 2
  maximum-idle-time = 300 # 5 minutes
  environment = {
    AOP_PAGE_HEIGHT -> 11i
    AOP_PAGE_WIDTH -> 8.5i
    AOP_RESOLUTION -> 300
    AOP_HORIZONTAL_MARGINS -> 0.167i
    AOP_VERTICAL_MARGINS -> 0.167i
  }
;
```

Editing the aopd.conf Configuration File

Add or edit this statement in the **aopd.conf** configuration file to start the Transform Manager. This is not required to use only the SAP to AFP transform. See “Appendix A. Infoprint Server Files” on page 131 for more information about this configuration file.

start-daemons = { xfd }

To start the Transform Manager daemon, add **xfd** to the values in this statement before issuing the **aopstart** command. Enclose the values in braces. By default, only the Printer Inventory Manager and the LPD start. This statement is required to start the Transform Manager daemon.

Default: start-daemons = {lpd}

Setting Environment Variables

The following environment variable affects the behavior of the Transform Manager. It is not required for the SAP to AFP transform. You can set this environment variable in the **/etc/profile** file.

AOPXFD_CONF

Full path name of the transform configuration file. If you created the transform configuration file in **/etc/Printsrv/aopxfd.conf**, you do not need to set this environment variable.

The following environment variable affects the behavior of the SAP to AFP transform. You can set this environment variable in the **/etc/profile** file.

AOP_SAP2AFP_RESOURCES

Path where resources used by the SAP to AFP transform are located. If you installed Infoprint Server Transforms resources in the default directory, **/usr/lpp/Printsrv/sap2afp/**, you do not need to set this environment variable.

Installing Multibyte Conversion Tables for SAP to AFP Transform

In order to transform and print SAP data streams that contain multibyte special characters, such as data streams produced by Asian versions of the R/3 system, you must install binary conversion tables for the code pages used by the SAP to AFP transform, **sap2afp**.

Use the **uconvdef** command to create the required conversion tables in the **/usr/lib/nls/locale/uconvTable** directory. You need root authority to install these tables. For more information about the **uconvdef** command, refer to *OS/390 UNIX System Services Command Reference*.

Infoprint Server Transforms provides the following source files to be used as input to the **uconvdef** command in the **/usr/lpp/Printsrv/sap2afp** directory. These source files define the mapping between UCS-2 and multibyte code sets:

IBM-932-SAP2AFP.ucmap

Defines the mapping for the SAP R/3 8000 (Japanese) system code page. The **sap2afp** transform uses code page IBM-932-SAP2AFP as the **iconv** source code page.

IBM-1031-SAP2AFP.ucmap

Defines the mapping for the IBM-1031-SAP2AFP code page, which the **sap2afp** transform uses as the default output code page. This mapping is based on IBM-290.ucmap, with the addition of the 11 box drawing characters. You can change the output code page in the **80000000.tab** configuration file, described in “xxxx0000.tab Configuration File” on page 59.

IBM-1030-SAP2AFP.ucmap

Defines the mapping for the IBM-1030-SAP2AFP code page. This mapping is based on IBM-1027.ucmap, with the addition of the 11 box drawing characters. To use the IBM-1030-SAP2AFP code page, customize the **80000000.tab** configuration file.

Create the conversion tables in the **/usr/lib/nls/locale/uconvTable** directory.

Customizing SAP to AFP Configuration Files

The SAP to AFP transform, **sap2afp**, uses the following configuration files:

- **barcode.tab**
- **defcp.tab**
- **fonts.tab**
- **image.tab**
- **pagedef.tab**
- **printer.tab**
- **xxxx0000.tab**

Infoprint Server Transforms provides default configuration files in the **/usr/lpp/Printsrv/sap2afp** directory. If you choose to put the SAP to AFP configuration files in a different directory, then name the new directory in environment variable **AOP_SAP2AFP_RESOURCES**. All of the configuration files must be in the same directory.

You can optionally customize the default SAP to AFP configuration files for your installation. If you edit the default files, the following considerations apply:

- All tables can contain comments (starting with **//** in column 1) and empty lines. These lines are ignored by the **sap2afp** transform.
- All table entries are case-sensitive.

barcode.tab Configuration File

The **barcode.tab** configuration file contains a table that maps bar code names used by SAP to the matching bar codes defined by the AFP Bar Code Object Content Architecture (BCOCA).

File **barcode.tab** uses the following keywords:

BarCode	Specifies the OTF bar code names (SAPBARCODE parameter of the BC OTF command). The maximum length of this field is 8 bytes. It can contain any value.
Type	Specifies the AFP bar code type as defined in <i>IBM Data Stream and Object Architectures: Bar Code Object Content Architecture Reference</i> for the Barcode Type parameter of the Barcode Data Descriptor structured field. This value can be any decimal value between 0 and 999. The sap2afp transform does not verify that this value maps to a valid hexadecimal barcode type.
Mode	Specifies the modifier byte within the Barcode Modifier parameter of the Barcode Data Descriptor structured field. Any numeric value between 0 and 999 is accepted and not verified.
Flag	Controls the printing of the Human Readable Interface (HRI) character. Specify one of the following values: 0 Causes the HRI character to be printed

128 Causes the **HRI** character not to be printed

Notes:

1. The system administrator is responsible for the values entered in the table. Invalid values are not verified and may result in errors.
2. The second group of bar codes in the **barcode.tab** file (following the blank line) is supported from R/3 Release 3.0A only.

Figure 12 on page 52 shows the default **barcode.tab** configuration file.

```

// Barcode table

// Format : Barcode=ARTNR Type=017 Mode=002

// BarCode = 30F9      Type = 001 Mode = 001 Flag = 000
// BarCode = 30F91     Type = 001 Mode = 002 Flag = 000
// BarCode = MSI       Type = 002 Mode = 002 Flag = 000
// BarCode = UPCA      Type = 003 Mode = 000 Flag = 000
// BarCode = UPCE      Type = 005 Mode = 000 Flag = 000
// BarCode = UPC2      Type = 006 Mode = 000 Flag = 000
// BarCode = UPC5      Type = 007 Mode = 000 Flag = 000
// BarCode = EAN8       Type = 008 Mode = 000 Flag = 000
// BarCode = EAN13     Type = 009 Mode = 000 Flag = 000
// BarCode = IND25     Type = 010 Mode = 002 Flag = 000
// BarCode = MAT25     Type = 011 Mode = 002 Flag = 000
// BarCode = INTER     Type = 012 Mode = 002 Flag = 000
// BarCode = CODA      Type = 013 Mode = 002 Flag = 000
// BarCode = C128      Type = 017 Mode = 002 Flag = 000
// BarCode = EAN2       Type = 022 Mode = 000 Flag = 000
// BarCode = EAN5       Type = 023 Mode = 000 Flag = 000
// BarCode = POST      Type = 024 Mode = 002 Flag = 000
// BarCode = AUFNR     Type = 001 Mode = 001 Flag = 128

BarCode = ARTNR      Type = 017 Mode = 002 Flag = 000
BarCode = AUFNR      Type = 017 Mode = 002 Flag = 000
BarCode = BARCLVS    Type = 001 Mode = 001 Flag = 000

BarCode = ARTNR      Type = 017 Mode = 002 Flag = 000
BarCode = AUFNR      Type = 017 Mode = 002 Flag = 000
BarCode = BARCLVS    Type = 001 Mode = 001 Flag = 000

BarCode = BC_C128B   Type = 017 Mode = 002 Flag = 128
BarCode = BC_CD39    Type = 001 Mode = 001 Flag = 128
BarCode = BC_CD39C   Type = 001 Mode = 002 Flag = 128
BarCode = BC_EAN13   Type = 009 Mode = 000 Flag = 128
BarCode = BC_EAN8    Type = 008 Mode = 000 Flag = 128
BarCode = BC_EANH    Type = 017 Mode = 002 Flag = 128
BarCode = BC_I25     Type = 012 Mode = 001 Flag = 128
BarCode = BC_I25C    Type = 012 Mode = 002 Flag = 128
BarCode = BC_MSI     Type = 002 Mode = 001 Flag = 128
BarCode = BC_MSIC    Type = 002 Mode = 002 Flag = 128
BarCode = BC_MSIC1   Type = 002 Mode = 003 Flag = 128
BarCode = BC_MSIC2   Type = 002 Mode = 005 Flag = 128
BarCode = BC_PSN5    Type = 024 Mode = 000 Flag = 128
BarCode = BC_PSN9    Type = 024 Mode = 001 Flag = 128

BarCode = KUNAUNR    Type = 017 Mode = 002 Flag = 000
BarCode = KUNAUPS    Type = 017 Mode = 002 Flag = 000
BarCode = MBBARC     Type = 017 Mode = 002 Flag = 000
BarCode = MBBARC1    Type = 008 Mode = 000 Flag = 000
BarCode = RSNUM      Type = 017 Mode = 002 Flag = 000
BarCode = RSPOS      Type = 017 Mode = 002 Flag = 000
BarCode = RUECKNR    Type = 017 Mode = 002 Flag = 000

```

Figure 12. Default **barcode.tab** Configuration File

defcp.tab Configuration File

The **defcp.tab** configuration file contains a table that maps the Open Systems EBCDIC 1047 code page to the code page associated with the ABAP coded fonts specified in the **pagedef.tab** file.

Figure 13 on page 53 shows the default **defcp.tab** file. The table in this file maps to output code page T1DBBASE for use with coded fonts GT2B, GT5B, GT8B, and GB2B, which are specified in the default **pagedef.tab** file.

The **sap2afp** transform maps the ABAP box characters to input code points 0-a. File **defcp.tab** maps these code points to the appropriate code points in the output code page. The mapping of box characters is described in the header of the default file shown in Figure 13.

```
// ABAP/Line Data Code Page
//
// Maps codepoints from codepage "Open Systems 1047" + box characters (0 - a)
//           to codepage "TIDBBASE (Belgium/Swiss/Luxembourg-DBBase)"
//
// NOTE: Codepage TIDBBASE chosen because it has box characters.
//
// ABAP box characters mapped to 0-a by sap2afp transform:
//  0 = upper left box corner
//  1 = lower left box corner
//  2 = upper right box corner
//  3 = lower right box corner
//  4 = center box bar horizontal
//  5 = center box bar vertical
//  6 = left middle box side
//  7 = right middle box side
//  8 = middle box bottom
//  9 = middle box top
//  a = box intersection

// Format : Ebcdic=Ebcdic

0 = ac
1 = ad
2 = ae
:
:

fd = fd
fe = fe
ff = 40
```

Figure 13. Default **defcp.tab** Configuration File

Asian versions of SAP R/3 use special coding to represent double-byte special characters; therefore, the **defcp.tab** file must specify the SAP R/3 system code page and the output code page conversion table. The following keywords in the table specify the input and output code pages:

CONVERT FOR=xxxx

Specifies the R/3 system code page, where xxxx is one of the following values:

8000	Japanese
8300	Chinese – traditional/Taiwan
8400	Chinese – simplified/PR China
8500	Korean

CONVERT TO=yyyyyyyyyy

Specifies the output code page conversion table, for example, IBM-1031-SAP2AFP.

Figure 14 on page 54 shows the Japanese conversion table in file **defcp.tab.japan**. To use this file, rename it to **defcp.tab**.

```
// defcp.tab
//-----
// This file is used internally by sap2afp, as well as to convert ABAP
// data to linedata.

// SAP2AFP is configured to use IBM-932 as the iconv "From" value. This table
// sets the "To" value to IBM-1031.

CONVERT FOR=8000
CONVERT TO=IBM-1031-SAP2AFP
```

Figure 14. Sample **defcp.tab.japan** Configuration File

fonts.tab Configuration File

The **fonts.tab** configuration file contains a table that maps the fonts used in the SAP OTF data stream to AFP fonts. The following font families are predefined with SAP R/3:

Font Family	Font
COURIER	Courier
HELVE	Helvetica
LETGOTH	Letter Gothic
LNPRINT	Line Print
TIMES	Times New Roman
OCRA	Optical Character Recognition A
OCRB	Optical Character Recognition B
JPMINCHO	Heisei Mincho
DBMINCHO	Heisei Mincho
DBGOTHIC	Heisei Gothic

Note: JPMINCHO and DBMINCHO are different names for the same font.

A font named **BARCODE** must be defined for the **HRI** character of a bar code.

The following keywords in the **fonts.tab** configuration file set the format of the fonts you use to print with R/3:

DefCodePage Specifies the default SBCS code page used if no **FC** OTF command is given or if the requested font is not found in the **fonts.tab** table.

DefCharSet Specifies the default SBCS character set used if no **FC** OTF command is given or if the requested font is not found in the **fonts.tab** table.

DBDefCodePage Specifies the default DBCS code page used if no **FC** OTF command is given or if the requested font is not found in the **fonts.tab** table.

DBDefCharSet Specifies the default DBCS character set used if no **FC** OTF command is given or if the requested font is not found in the **fonts.tab** table.

SBDefCodePage Specifies the default SBCS code page used for half-width characters in DBCS fonts if no **FC** OTF command is given or if the requested font is not found in the **fonts.tab** table.

SBDefCharSet Specifies the default SBCS character set used for half-width

	characters in DBCS fonts if no FC OTF command is given or if the requested font is not found in the fonts.tab table.
Font	Describes the font family (FONTFAMILY parameter of the FC OTF command). The maximum size is 8 bytes and content is not verified.
Size	Specifies the font size in 1/10 of a point (FONT SIZE parameter of the FC OTF command). The value must be numeric and is not verified.
Type	Defines the font type (BOLD and ITALIC parameter of the FC OTF command). Type=0 is normal, Type=1 is italic, Type=2 is bold, and Type=3 is italic bold. Any other value is invalid.
CodePage	Specifies the code page and requires a valid AFP code page name (8 bytes). The value is not verified. An invalid name can result in an error message.
CharSet	Specifies the AFP font character set. The content is not verified.
CodedFont	Specifies the AFP coded font. The content is not verified.
DB	Specifies whether the font is an SBCS font (DB=0) or a DBCS font (DB=1).
SBCodePage	Specifies the single-byte code page used for half-width characters in DBCS fonts. The value is not verified. An invalid name can result in an error message.
SBCharSet	Specifies the AFP single-byte font character set used for half-width characters in DBCS fonts. The content is not verified.
SBCodedFont	Specifies the AFP single-byte coded font used for half-width characters in DBCS fonts. The content is not verified.

If an SBCS font that matches the **Font**, **Size**, and **Type** values is not found, the code page and font character set from the **DefCodePage** and **DefCharSet** keywords are used and a warning message is displayed. If a DBCS font that matches the **Font**, **Size**, and **Type** values is not found, the code pages and font character sets from the **DBDefCodePage**, **SBDefCodePage**, **DBDefCharSet**, and **SBDefCharSet** keywords are used and a warning message is displayed.

The SAP device types **IBMAFP** and **IBMAFP3** support the predefined font families. These font families are also supported as IBM Expanded Core Fonts, or IBM Chinese, Japanese, and Korean (CJK) Fonts.

For ABAP listings, use the Letter Gothic Latin1 font provided with the AFP Font Collection V2 (program number 5648-B33). If you choose not to install the AFP Font Collection and plan to use the Gothic Text fonts provided with the IBM Compatibility Fonts, adjust the font names in the **pagedef.tab** configuration file.

Figure 15 on page 56 and Figure 16 on page 56 show sections of the **fonts.tab** configuration file.

```
// Fonts table

// Format : Font=COURIER Size=070 Type=0 CodePage=T1V10500 CharSet=C0420070
//          DefCodePage = T1V10500
//          DefCharSet = C0420000

DefCodePage = T1V10500
DefCharSet = C0420000
DBDefCodePage = T10300
DBDefCharSet = CZJHMN
SBDefCodePage = T1H01027
SBDefCharSet = CZJHMN

// Courier
Font=COURIER Size=070 Type=0 CodePage=T1V10500 CharSet=C0420070 DB=0
Font=COURIER Size=070 Type=1 CodePage=T1V10500 CharSet=C0430070 DB=0
Font=COURIER Size=070 Type=2 CodePage=T1V10500 CharSet=C0440070 DB=0
Font=COURIER Size=070 Type=3 CodePage=T1V10500 CharSet=C0450070 DB=0
Font=COURIER Size=080 Type=0 CodePage=T1V10500 CharSet=C0420080 DB=0
Font=COURIER Size=080 Type=1 CodePage=T1V10500 CharSet=C0430080 DB=0
Font=COURIER Size=080 Type=2 CodePage=T1V10500 CharSet=C0440080 DB=0
Font=COURIER Size=080 Type=3 CodePage=T1V10500 CharSet=C0450080 DB=0
Font=COURIER Size=100 Type=0 CodePage=T1V10500 CharSet=C0420000 DB=0
Font=COURIER Size=100 Type=1 CodePage=T1V10500 CharSet=C0430000 DB=0
Font=COURIER Size=100 Type=2 CodePage=T1V10500 CharSet=C0440000 DB=0
Font=COURIER Size=100 Type=3 CodePage=T1V10500 CharSet=C0450000 DB=0
Font=COURIER Size=120 Type=0 CodePage=T1V10500 CharSet=C04200B0 DB=0
Font=COURIER Size=120 Type=1 CodePage=T1V10500 CharSet=C04300B0 DB=0
Font=COURIER Size=120 Type=2 CodePage=T1V10500 CharSet=C04400B0 DB=0
Font=COURIER Size=120 Type=3 CodePage=T1V10500 CharSet=C04500B0 DB=0
Font=COURIER Size=140 Type=0 CodePage=T1V10500 CharSet=C04200D0 DB=0
Font=COURIER Size=140 Type=1 CodePage=T1V10500 CharSet=C04300D0 DB=0
Font=COURIER Size=140 Type=2 CodePage=T1V10500 CharSet=C04400D0 DB=0
Font=COURIER Size=140 Type=3 CodePage=T1V10500 CharSet=C04500D0 DB=0
Font=COURIER Size=200 Type=0 CodePage=T1V10500 CharSet=C04200J0 DB=0
Font=COURIER Size=200 Type=1 CodePage=T1V10500 CharSet=C04300J0 DB=0
Font=COURIER Size=200 Type=2 CodePage=T1V10500 CharSet=C04400J0 DB=0
Font=COURIER Size=200 Type=3 CodePage=T1V10500 CharSet=C04500J0 DB=0
```

Figure 15. Header and Courier Portions of Default **fonts.tab** Configuration File

```
// Japanese Gothic
Font=DBGOTHIC Size=060 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027 SBCharSet=CZJHKG
Font=DBGOTHIC Size=080 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027 SBCharSet=CZJHKG
Font=DBGOTHIC Size=100 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027 SBCharSet=CZJHKG
Font=DBGOTHIC Size=120 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027 SBCharSet=CZJHKG
Font=DBGOTHIC Size=140 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027 SBCharSet=CZJHKG
Font=DBGOTHIC Size=160 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027 SBCharSet=CZJHKG
Font=DBGOTHIC Size=180 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027 SBCharSet=CZJHKG
```

Figure 16. Heisei Gothic Portion of Default **fonts.tab** Configuration File

image.tab Configuration File

The **image.tab** configuration file contains a table that defines values used to print image data. It contains the following parameters:

DEFRES	Specifies the default resolution used for printing image data if you do not specify a resolution in the -r option on lp or sap2afp command. Valid values are 240 , 300 , 480 , and 600 .
Width	Specifies the width of the dither matrix.
Height	Specifies the height of the dither matrix.

Cell	Specifies the values for the dither matrix.
Transform	Specifies 256 grayscale correction values.

Infoprint Server Transforms provides the following **image.tab** files. To use one of these files, rename it to **image.tab**:

- **image.tab** - same as **image.tab.85lr**
- **image.tab.141ap** - 141 lines/inch AppleWriter grayscale emulation
- **image.tab.141dt** - 141 lines/inch Xerox DocuTech grayscale emulation
- **image.tab.141lr** - 141 lines/inch linear gamma correction
- **image.tab.85ap** - 85 lines/inch AppleWriter grayscale emulation
- **image.tab.85dt** - 85 lines/inch Xerox DocuTech grayscale emulation
- **image.tab.85lr** - 85 lines/inch linear gamma correction

Figure 17 shows the default **image.tab** configuration file.

```
// image.tab
//
// This file is used to determine the dither matrix and the
// grayscale correction values.

// Define the default output resolution
DEFRES = 600

// Define the dither matrix
Width = 78
Height = 78
Cell =
  15  63 184 219 249 240 158 109  98  54  82 118 133 113  75  64
  87 104 166 215 234 229 179  62  38   4  15  59 186 220 246 240
 166 123  87  61  65 108 133 118  78  64 100 105 167 219 234 230
  :
 28  73 193 223 253 247 199  76  31   8  20  93 148 156 203 178
136 126 144 186 201 155 151  90  43  24  28  75 199 222 254 251
196  79  31   7

//
// Grayscale correction values
//

Transform =
0
1
2
:
248
252
255
```

Figure 17. Default **image.tab** Configuration File

pagedef.tab Configuration File

The **pagedef.tab** configuration file contains a table that maps the *PJPAPEr parameter (a parameter that is added to the header of the SAP file with the L or Z access methods) to page definition and form definition names. File **pagedef.tab** also defines the fonts to be used for ABAP data.

File **pagedef.tab** contains the following parameters:

Paper	Specifies the value of the OTF print option parameter PJPAPER . The value is not verified.
FormDef	Specifies the name of the form definition to be used for printing both OTF and ABAP reports. This value is not verified; however, an error results if PSF cannot find the form definition during printing.
PageDef	Specifies the name of the page definition to be used for printing ABAP reports. This value is not verified; however, an error results if PSF cannot find the page definition during printing.
FontNorm	Specifies the normal coded font used for line data printing. The value is not verified.
FontBold	Specifies the bold coded font used for line data printing. The value is not verified.

Figure 18 on page 59 shows the default **pagedef.tab** configuration file.

```

// PageDef table
//-----
// This table allows users to configure resource selection based on a
// Format in the SAP system. Thus a format, say ZCHECK, could be set-up
// to use a Formdef that would invoke an Overlay for checks.
//
// Although all paper types are defined in the same format, conventions
// on the SAP system should be observed as follows:
// + ABAP user defined formats should be in the form Z_ROWS_COLUMNS or
//   Z_ROWS_COLUMNS_EXTRA-ID. An example of a user defined ABAP format
//   would be Z_65_132_MYABAP.
// + Formats for OTF in the SAP system are formats listed that do not
//   start with a X (i.e. LEGAL). An example of a user defined OTF format
//   would be ZMYFORMAT.
// + Do NOT copy ABAP formats as OTF formats or OTF formats as ABAP formats!
//
// Resource selection is done as follows:
// + For ABAP data, all entries (FormDef, PageDef, FontNorm, FontBold)
//   are used.
// + For OTF data, only the FormDef value is used. However, the other
//   field values must be defined.
//-----
// Format: Paper=X_65_132 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
// Notes:
// =====
// 1) the pagedefs apply to letter size paper.
//    A4 paper size may require modifying the PageDef field.
// 2) Z_INCH12 is a sample for a user specification
// 3) A format (Paper) must not exceed 16 characters.
//-----

// ABAP Formats
Paper=X_65_132 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=X_44_120 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=X_58_170 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT8B FontBold=GT8B
Paper=X_65_255 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GT2B
Paper=X_65_80 FormDef=F1A10111 PageDef=P1C09182 FontNorm=GT2B FontBold=GB2B
Paper=X_90_120 FormDef=F1A10111 PageDef=P1C09182 FontNorm=GT5B FontBold=GT5B
Paper=X_PAPER FormDef=F1A10111 PageDef=P1C09182 FontNorm=GT2B FontBold=GB2B

// OTF Formats
Paper=DINA3 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=DINA4 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=DINA5 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=EXECUTIV FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=INCH4 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=INCH6 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=INCH7 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=INCH8 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=INCH11 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=INCH12 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=LEGAL FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=LETTER FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=LINE_21 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B
Paper=LINE_22 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2B FontBold=GB2B

//Sample of a added user-defined pagedef entry:
//Paper=Z_INCH12 FormDef=F1A10111 PageDef=P1SAPPD FontNorm=GT2B FontBold=GB2B

```

Figure 18. Default **pagedef.tab** Configuration File

xxxx0000.tab Configuration File

The following configuration files contain tables that map characters of an individual SAP code page into an AFP code page.

Table 4. SAP Code Page Configuration Files

File Name	Input Code Page	Description	Output Code Page	Description
00000000.tab	T1000819	Latin-1 ISO ANSI 8-bit (ISO8859-1)	T1V10500	International #5
01200000.tab	T1V10500	International #5	T1V10500	International #5
11000000.tab	T1000819	Latin-1 ISO ANSI 8-bit (ISO8859-1)	T1V10500	International #5
11000000.tab.japan ¹	T1000819	Latin-1 ISO ANSI 8-bit (ISO8859-1)	T1V10281	Japan (Latin)-CECP
40010000.tab	T1000876 ²	OCR-A ASCII	T1000892	OCR-A
40020000.tab	T1000877 ²	OCR-B ASCII	T1000893	OCR-B
80000000.tab	IBM-932-SAP2AFP	Japanese	IBM-1031-SAP2AFP	Japanese

1. To produce output in the T1V10281 Japan (Latin) code page, rename file **11000000.tab.japan** to **11000000.tab**.
2. The input OCR-A and the OCR-B code pages are equivalent to ISO-8859-1 with the special characters **hook**, **fork**, and **chair**.

Asian versions of SAP R/3 use special coding to represent double-byte special characters. Therefore, the **xxxx0000.tab** file must indicate the R/3 system code page and the output code page. The following parameters in the table specify the input and output code pages:

CONVERT FOR=xxxx

Specifies the R/3 system code page, where xxxx is one of the following values:

8000 Japanese
8300 Chinese – traditional/Taiwan
8400 Chinese – simplified/PR China
8500 Korean

CONVERT TO=yyyyyyyyyy

Specifies the output code page conversion table, for example IBM-1031-SAP2AFP.

Figure 19 shows the default **80000000.tab** configuration file used for the R/3 system code page 8000 (Japanese).

```
// Japanese
CONVERT FOR=8000
CONVERT TO=IBM-1031-SAP2AFP
```

Figure 19. Default **80000000.tab** Configuration File

Chapter 6. Customizing the SNMP Subagent

The Infoprint Server SNMP subagent, in conjunction with support provided by PSF for OS/390¹ and the OS/390 SNMP agent, lets your installation monitor printer characteristics and printer status for printers controlled by PSF for OS/390 that are not TCP/IP-attached or do not have internal agents. Your installation must use an SNMP manager application, such as IBM Network Printer Manager for the Web, to monitor printer information. See “SNMP Subagent” on page 12 for an overview of the SNMP subagent and how it fits into your OS/390 system.

This chapter describes the following required and optional tasks to customize and the SNMP subagent:

Task	Condition	See Page:
Customizing and Starting the OS/390 SNMP Agent	Required	61
Editing the aopd.conf Configuration File	Required	62
Enabling SNMP Reporting in the Printer Inventory	Required	62
Customizing the SNMP Manager	Required	63

After customizing the SNMP subagent, the following tasks must be performed; refer to *OS/390 Infoprint Server Operation and Administration* for information about starting the SNMP agent and subagent.

- Start the OS/390 SNMP agent if not already started.
- Start the SNMP subagent with the **aopstart** command or AOPSTART procedure.
- Start each PSF printer at least once after enabling SNMP reporting in the Printer Inventory and before defining PSF printers to an SNMP manager.

Customizing and Starting the OS/390 SNMP Agent

You must configure and start the OS/390 SNMP agent before starting the Infoprint Server SNMP subagent. To configure and start the OS/390 SNMP agent, refer to *OS/390 SecureWay Communications Server: IP Configuration*.

Following are some considerations related to configuring the OS/390 SNMP agent:

- The community-name function of SNMP can provide a higher-level of security for your installation; however, IBM does not recommend using the community-name function for viewing information about PSF printers because administrators cannot change printer characteristics. If your installation wants to use the community-name function, then you must ensure that the same community name is specified (1) to the OS/390 SNMP agent, (2) in the **snmp-community** statement in the **aopd.conf** configuration file, and (3) to the SNMP manager. Also, when you select a community-name, be aware that the SNMP manager you use might restrict the length of the community name or the allowed characters.
- Be sure to provide SNMPTRAP.DEST information to define the IP addresses of the SNMP managers to which the OS/390 SNMP agent sends traps. When you

1. PSF for OS/390 3.1 with APAR OW37851, or a later release of PSF, is required.

provide trap information, the OS/390 SNMP agent can notify the SNMP manager of printer alert conditions more quickly.

- Do not reserve ports 161 and 162 for the OS/390 SNMP agent in the PROFILE.TCPIP data set. See “Customizing TCP/IP” on page 28 for more information.

Editing the aopd.conf Configuration File

You must customize the Infoprint Server **aopd.conf** configuration file before starting the SNMP subagent. Add or edit these statements in the configuration file. See “Appendix A. Infoprint Server Files” on page 131 for more information about the configuration file.

snmp-community = name

The name assigned to the SNMP community for making SNMP requests to the OS/390 system. If you specified a community name to the OS/390 SNMP agent, specify the same community name in this statement. If you did not provide a community name to the OS/390 SNMP agent, the OS/390 SNMP agent accepts requests from any SNMP manager with the community name of **public**.

Specify 1–32 uppercase or lowercase letters, numbers, or special characters; however, note that the OS/390 SNMP agent and the SNMP manager might restrict the length of the community name or the allowed characters. The community name is case sensitive.

Default: snmp-community = public

start-daemons = {snmpd}

To start the SNMP subagent with the **aopstart** command, add **snmpd** to the values in this statement. Enclose the values in braces. By default, only the Printer Inventory Manager and the LPD start; therefore, this statement is required to start the SNMP subagent.

Default: start-daemons = {lpd}

Enabling SNMP Reporting in the Printer Inventory

The administrator can enable SNMP reporting for a PSF printer in the FSA definition for the PSF printer in the Printer Inventory. By default, SNMP reporting is disabled. To enable SNMP reporting, the administrator selects the **SNMP reporting** field on the ISPF panel used to add or edit an FSA definition.

If a TCP/IP-connected printer contains an internal SNMP agent, IBM recommends that you define the printer directly to the SNMP manager and not enable SNMP reporting in the FSA definition. When you define a printer directly to the SNMP manager, the administrator can also view printer statistics and change some printer characteristics. Refer to the documentation for your TCP/IP-enabled printers to determine if they have internal SNMP agents.

The administrator should enable SNMP reporting in the Printer Inventory and also start the PSF printers before defining the PSF printers to the SNMP manager.

Refer to *PSF for OS/390: Customization* for information about how to customize PSF to use information in the Printer Inventory and enable SNMP reporting. Refer to *OS/390 Infoprint Server Operation and Administration* for information about how to add or edit an FSA definition.

Customizing the SNMP Manager

To monitor PSF printers, you must install an SNMP manager. You can install any SNMP manager that communicates with an SNMP agent that supports the general printer MIB (defined in RFC 1759) and has implemented support for multiple printers defined with one IP address (the IP address of the OS/390 system).

IBM provides an SNMP manager, IBM Network Printer Manager (NPM) for the Web, which contains support to display the status of print systems (such as an OS/390 system) that contain multiple printers. When you use NPM, administrators can use a Java-enabled version of Netscape Navigator or Microsoft Internet Explorer to monitor printers; the NPM server runs on a Windows NT system. NPM for the Web limits the number of printers you can monitor within each print system; see the NPM online help for the maximum number of printers NPM allows.

You can download the latest version of NPM for the Web from the IBM Printing Systems Company Web site: <http://www.ibm.com/printers>. Use the NPM README file and online help for information about how to install and use NPM.

Following are some considerations related to customizing NPM for the Web:

- To add the OS/390 print system to NPM, you need to know the IP address (in dotted-decimal notation) of the OS/390 host and the community name that you defined to the OS/390 SNMP agent and to the SNMP subagent. If you have not defined a community name, use the default **public**.
- Before you add the OS/390 print system to NPM, you must (1) start the OS/390 SNMP agent and SNMP subagent, (2) enable SNMP reporting for PSF printers in the Printer Inventory, and (3) start or restart the PSF printers for which you enabled SNMP reporting.

Any PSF printer can be inactive when you define the OS/390 print system, provided the printer has been started at least once after SNMP reporting was enabled in the Printer Inventory. If you do not start *any* PSF printers after enabling SNMP reporting, NPM displays a status of NOT SUPPORTED. If this situation occurs, cycle the NPM server.

- When you add the OS/390 print system to NPM, NPM automatically discovers each PSF printer for which SNMP reporting is enabled in the Printer Inventory. NPM lists the FSA definition name for the PSF printers it discovers. If a PSF printer is not in the list, start or restart the PSF printer and cycle the NPM server.
- If you enable SNMP reporting for a PSF printer in the Printer Inventory after adding the OS/390 print system to NPM, start or restart the PSF printer and cycle the NPM server.
- If you disable SNMP reporting for a PSF printer in the Printer Inventory after adding the OS/390 print system to NPM, cycle the NPM server.

Chapter 7. Customizing NetSpool

The NetSpool component of Infoprint Server automatically intercepts VTAM print requests from VTAM applications, such as CICS and IMS, and allocates output data sets on the JES spool. See “NetSpool” on page 8 for an overview of NetSpool and how this component fits into your system.

This chapter describes the following required and optional tasks to customize NetSpool:

Task	Condition	See Page:
Allocating and Initializing the Message Log	Optional: To record messages in a log	65
Creating a NetSpool Startup Procedure	Required	66
Writing NetSpool Exit Programs	Optional: To add, modify, or delete transparent data in SCS data	69

After customizing NetSpool, the following tasks must be performed; refer to *OS/390 Infoprint Server Operation and Administration* for information about these tasks:

- Create a printer definition for each target printer in the Printer Inventory and specify printer attributes required by NetSpool.
- Define each NetSpool printer logical unit (LU) to VTAM; modify VTAM resource definitions if necessary.
- Start NetSpool and vary the printer LUs active.

Allocating and Initializing the Message Log

NetSpool can write informational and error messages in a NetSpool message-log data set. The NetSpool message-log data set is an optional data set that is organized as a circular queue. When a message extends to the end of the data set, the next message starts at the beginning. In this way, the size of the message data set remains stable, with old messages automatically overlaid by newer messages.

To use the NetSpool message log, you must allocate space for the data set and initialize it. You can use the sample JCL in SYS1.SAMPLIB (APIMIJCL) to allocate space and run the ANFMFILE program to initialize the data set.

The NetSpool message-log data set requires the following specifications:

- Sequential organization (PS)
- Fixed block format (FBA)
- Logical record length (LRECL) of 57 at a minimum; 80 is recommended
- Block size (BLKSIZE) of about 10 records per block; 800 is recommended

Secondary space is unnecessary, because the ANFMFILE program formats all available space. To change the size of the NetSpool message-log data set, rerun the ANFMFILE program.

Note: The ANFMFILE program produces a D37 abend when it completes processing.

Creating a NetSpool Startup Procedure

Before starting NetSpool, you must create a startup procedure. The startup procedure identifies:

- Program name, region size, time-out value, classes of printer LUs, tracing option, console name, and the name of the Printer Inventory
- The location of the message-log data set
- The location of the trace options file
- The location of exit programs

Note: If you change the NetSpool startup procedure, you must restart NetSpool to pick up the changes.

Determining the Number of NetSpool Programs to Start

You can run multiple instances of NetSpool simultaneously, each instance running in a different address space. The following reasons explain why this might be useful:

- To spread processing across multiple address spaces
- To reduce region-size requirements by spreading the printer LUs across multiple regions
- To enable different sets of printer LUs to be started and stopped at different times

Using Japanese or Spanish National Language Versions

Japanese and Spanish National Language Versions (NLVs) are available for NetSpool. If you ordered the Japanese or Spanish NLV, a new message data set is provided in a separate FMID. To install the message data set associated with the language feature, refer to *OS/390 Program Directory*.

To receive messages in a language other than English, specify the LANGUAGE parameter in the EXEC statement, using the last three characters in the name of the message data set. See “JCL Parameters” on page 67 for more information about the LANGUAGE parameter.

If the language feature is installed, you can still use the English message data set by either removing the LANGUAGE parameter from the EXEC statement or by specifying LANGUAGE=ENU and starting NetSpool again.

Be certain your console supports the language you install. NetSpool issues all messages to both the console and an optional message-log data set. Some NetSpool messages are issued before NetSpool loads the message data set. These messages will always be issued in English.

You must configure an extended MCS console on TSO/E to view translated messages. Refer to *OS/390 TSO/E Customization* for information about customizing TSO/E. Messages issued by NetSpool are already translated into the appropriate language.

Using an Extended MCS Console

When using extended MCS consoles, messages issued in response to NetSpool operator commands are routed to the console that issued the command. NetSpool

sends unsolicited messages to an extended MCS console when the CONSNAME parameter in the startup procedure specifies the name of an extended MCS console.

NetSpool Startup Procedure

NetSpool provides a startup procedure in SYS1.PROCLIB (APIJPJCL), which is shown in Figure 20. You can copy and modify this procedure for your installation. This procedure use symbolic parameters with default values for some of the parameters that you might want to change for your installation. Refer to *OS/390 MVS JCL Reference* for more information on symbolic parameters.

```
//NETSPOOL PROC LUCLASS=(1),INV='AOP1',DURATION=NOLIMIT,
//          SIZE=31M,MSGFILE=USER.MSGFILE
//APIPPAAA EXEC PGM=APIPPAAA,REGION=&SIZE,TIME=&DURATION, X
//          PARM='LUCLASS=&LUCLASS,INV=&INV'
/* Modify the next STEPLIB to replace the transparent data exit or
/* the beginning of file exit. This library must be APF
/* authorized.
/*STEPLIB DD DSN=NETSPOOL.EXIT.SLOADLIB,DISP=SHR
//APIMSG DD DSN=&MSGFILE,DISP=SHR
//SYSPRINT DD SYSOUT=*
```

Figure 20. NetSpool Startup Procedure — SYS1.PROCLIB(APIJPJCL)

JCL Parameters

This section describes the syntax of the JCL statements and parameters that you can include in your NetSpool startup procedure.

label PROC

Marks the beginning of the procedure. The sample PROC statement in Figure 20 specifies default values for symbolic parameters coded in the sample. If you plan to use the sample procedure, modify these default values to suit your installation.

label EXEC PGM=APIPPAAA,REGION=*size***,TIME=NOLIMIT,**
PARM='LUCLASS=(*class***[,...]),INV=***inventory***[,LANGUAGE=***language_id***] [**
,CONSNAME=*console_name***][,TRACE=ON]'**

Invokes the NetSpool program.

PGM=APIPPAAA

The NetSpool program entry point.

REGION=*size*

The maximum amount of virtual storage NetSpool can use. Refer to *OS/390 MVS JCL Reference* for more information about specifying region size.

TIME=NOLIMIT

Specify TIME=1440 or TIME=NOLIMIT to prevent TIMEOUT abends.

PARM=

Specify a maximum of 100 characters for the value of the PARM parameter, as described in *OS/390 MVS JCL Reference*.

LUCLASS=(*class***[,...])**

The classes of printer LUs that this instance of NetSpool is to start. The class of each printer LU is specified in the printer definition in the Printer Inventory.

Valid class values are 1 to 64. Enclose the classes in parentheses, even if you specify only one class value. To specify more than one class, separate each class with a comma. NetSpool starts all printer LUs that are assigned to any one of the classes specified. For example, if you specify `LUCLASS=(1,2)`, NetSpool starts all printer LUs assigned to class 1 and all printer LUs assigned to class 2. Refer to *OS/390 Infoprint Server Operation and Administration* for more information on using LU classes.

If you want to start NetSpool without starting any printer LUs, specify a class number, from 1 to 64, that does not match any classes defined for printer LUs in the Printer Inventory. To start individual printers, use the NetSpool LUNAME ADD command described in *OS/390 Infoprint Server Operation and Administration*.

This parameter is required.

INV=inventory

The name that you assigned to the Printer Inventory in the **inventory** statement in the **aopd.conf** configuration file. Specify AOP1 (the default name) if you have not created an **aopd.conf** configuration file or if the **inventory** statement is omitted. This name is case sensitive. This parameter is required.

CONSNAME=console_name

The name of an MCS or extended MCS console where you want NetSpool to display its unsolicited messages. Unsolicited messages are issued by NetSpool as a result of an unexpected error, such as a VTAM or data error. For *console_name*, specify a name that is defined in a CONSOLE statement in the CONSOLxx member of SYS1.PARMLIB. You can abbreviate the CONSNAME parameter as follows: **CON=console_name**.

Translated NetSpool messages can be displayed correctly when CONSNAME specifies the name of an extended MCS console that is configured for the language of the translated messages.

If the CONSNAME parameter is not specified, unsolicited NetSpool messages are displayed at the console that issued the START NetSpool command to start NetSpool.

LANGUAGE=language_id

The language that NetSpool is to use for messages.

Note: You can abbreviate this parameter as follows: **LNG=language_id**

NetSpool appends the *language_id* to the prefix APIMT to determine the name of the message data set to load. If you specify a language other than English, first install the NLV. If you omit the LANGUAGE parameter, or if NetSpool does not find the specified message data set, NetSpool issues messages in English. (NetSpool uses the default message data set APIMTENU.)

Example:

LNG=JPN

TRACE=ON

Specifies that NetSpool is to read the data set in the TRACEOPT DD statement to obtain the trace options. The tracing will start when NetSpool is started.

STEPLIB DD DSN=*dsname*,DISP=SHR

The name of a data set that contains NetSpool exit programs written by your installation. The library must be APF authorized. This statement is optional; specify it if you are replacing the IBM-supplied APPIPTD1 or APPIPTD2 exit programs.

APIMMSG DD DSN=*dsname*,DISP=SHR

The name of the NetSpool message-log data set. This statement is optional. If you include this statement, NetSpool writes messages to the message-log data set as well as to the operator console. If you specify a language identifier other than ENU in the LANGUAGE parameter on the NetSpool startup procedure, the message-log data set contains the translated messages. See “Allocating and Initializing the Message Log” on page 65 for more information.

TRACEOPT DD DSN=*dsname*,DISP=SHR

The name of the trace options data set. Specify DISP=SHR to allow more than one instance of NetSpool to use the data set. The trace options data set can be either a member of a PDS or a sequential data set.

Writing NetSpool Exit Programs

Note

This section contains programming-interface information.

NetSpool provides two exits that allow you to customize the processing of SCS data streams:

- The Beginning of File exit (APIPTD1)
This exit is called when NetSpool creates a new output data set.
- The Transparent Data Control exit (APIPTD2)
This exit is called when NetSpool finds the Transparent Data control (TRN) in the input data stream.

Note: These exits are called only when NetSpool is processing an SCS data stream.

Programming Considerations

Consider the following points when coding exit programs:

- You should be an experienced programmer and aware of overall system implications.
- Do not issue an explicit or implied WAIT in an exit, because it puts NetSpool into a WAIT state.
- The exits can run in either 24-bit or 31-bit addressing mode. 31-bit addressing is recommended, to allow more printer LUs to run in an address space.
- Use only the control block passed to the exit. Code references to fields in the control block using the names associated with each field in the DSECT control block. Do not use absolute displacements, because these displacements can change from one release to another.
- Code the exit to be reentrant for improved performance; however, if you replace an exit, you must stop and restart NetSpool to pick up the changed exit. If the exit is non-reentrant, you can replace the exit while NetSpool is running.
- The exit programs run in supervisor state in key 8.

- An ESPIE is in effect while the exit programs are running. Programming exceptions cause the printer session and the application session to terminate.

Note: NetSpool exits that run on OS/390 releases prior to V2R8 do not need to be changed or recompiled; however, the exits must be relinked with the AC(1) attribute and placed in an APF-authorized library. Future NetSpool program maintenance may require that you reassemble these exits.

APIPP1X2 Control Block

When the exits are called, register 1 contains the address of the APIPP1X2control block, mapped by DSECT S2STG, shown in Figure 21 on page 71.

Offset	(Hex)	Type	Length	Name	Description
0	(0)	Structure	*	S2STG	Exit parameter area
0	(0)	Character	5	S2EYEC	Set to 'S2STG'
5	(5)		1	*	Reserved
=====					
The following fields provide input to both exits. The exits must not change these fields.					
=====					
Horizontal Page Format:					
6	(6)	Signed	2	S2COL	Next column number in output
8	(8)	Signed	2	S2LM	Left margin column number
10	(0A)	Signed	2	S2RM	Right margin column number
12	(0C)	Signed	2	S2MPP	Maximum presentation position (line length)
Vertical Page Format:					
14	(0E)	Signed	2	S2LINE	Next output line number
16	(10)	Signed	2	S2TM	Top margin line number (channel 1)
18	(12)	Signed	22	S2C2T12	Channel 2-12 line numbers, 2 bytes per channel
40	(28)	Signed	2	S2BM	Bottom margin line number
42	(2A)	Signed	2	S2MPL	Maximum page length
44	(2C)	Character	8	S2PLUNAM	LU name of the application generating the print request (PLU)
52	(34)	Character	8	S2LPTNAM	Name of the printer LU (SLU)
60	(3C)	Character	8	S2PRTCLS	Classes of the printer LU.
					Each bit represents a class, from 1 to 64:
					Value Class
		1...			1 Class 1
		.1..			1 Class 2
	1			1 Class 8
					This pattern repeats for 7 more bytes.
					More than one bit can be set at a time.
=====					
The following field can be set by exit APIPPTD1.					
=====					
68	(44)	Character	1	S2DBCSM	Number of bytes to allocate per DBCS character:
					Value Meaning
					0 Two positions per character (default)
					1 One position per character
69	(45)		1	*	Reserved
=====					
The following fields can be set by both exits.					
=====					
70	(46)	Signed	2	S2NLNUM	New line number for output
72	(48)	Signed	2	S2NCNUM	New column number for output
=====					
The following field provides input to exit APIPPTD2. The exit must not change this field.					
=====					
76	(4C)	Character	275	S2TPDATA	Transparent data found in input data:
					1 byte: X'35'
					1 byte: Length of data
					* bytes: Transparent data
351	(15F)		5	*	Reserved
=====					
The following fields can be set by both exits.					
=====					
356	(164)	Signed	4	S2WRKLEN	Length of data in S2WRKARA
360	(168)	Character	275	S2WRKARA	Data to include in output

Figure 21. Format of APIPP1X2 Control Block

Beginning of File Exit (APIPPTD1)

The Beginning of File exit (APIPPTD1) receives control when NetSpool starts a new output data set, for SCS input data streams only.

Using this exit, you can add the TRN control and transparent data to the start of all or some data sets. You can also add other SCS controls, in addition to the TRN control. The SCS controls supported by NetSpool are described in “Appendix C. SCS Printer Simulation” on page 143. The total number of bytes you can add to the output data set is 275. NetSpool does not call the Transparent Data Control exit (APIPPTD2) for transparent data you add using this exit.

In order to format lines and pages, NetSpool keeps track of the next output print position, that is, the column number and line number at which the next character is to print. For transparent data, NetSpool increases the column and line number by the length of the transparent data, as specified in the TRN control.

The APIPPTD1 exit allows you to change the column and line number of the next print position *after* the transparent data. To assist in calculating the desired line and column numbers, NetSpool provides the exit with the column and line number of the next print position, *before* any transparent data is added. NetSpool also provides the exit with other page-formatting values.

Input to APIPPTD1

At input to the Beginning of File exit (APIPPTD1), the contents of the registers are:

Register 1	Address of the APIPP1X2 control block
Register 13	Address of an 18-word save area for saving the caller's registers
Register 14	Return address

The APIPP1X2 control block is described in Figure 21 on page 71. The following fields are valid at entry:

S2COL	The column number of the next print position
S2LM	Left margin column number
S2RM	Right margin column number
S2MPP	Maximum presentation position (line length)
S2LINE	The line number of the next print position
S2TM	Top margin line number (channel 1)
S2C2T12	Channel 2-12 line numbers
S2BM	Bottom margin line number
S2MPL	Maximum page length
S2PLUNAM	LU name of the application generating the print request (PLU)
S2LPTNAM	Name of the printer LU that NetSpool is processing (SLU)
S2PRTCLS	Logical-unit classes to which the printer LU belongs. Each bit in this field represents a class, with the high order-bit of the leftmost byte representing class 1. See Figure 21 on page 71 for a description of the bits.

Output From APIPPTD1

At return from the Beginning of File exit (APIPPTD1), restore the contents of all registers, and place one of the following codes in register 15 to tell NetSpool what action to take:

Code	Meaning
------	---------

- | | |
|---|---|
| 1 | Add the data from S2WRKARA to the output data set. |
| 2 | Do <i>not</i> add data from S2WRKARA to the output data set. |
| 4 | Close the VTAM session for this printer LU. |
| 8 | Do <i>not</i> call exit APIPPTD2 for any transparent data found in the data stream for this output data set. Do <i>not</i> add data from S2WRKARA to the output data set. |
| 9 | Do <i>not</i> call exit APIPPTD2 for any transparent data found in the data stream for this output data set. Add the data from S2WRKARA to the output data set. |

Also, set the following fields in the APIPP1X2 control block.

S2DBCSM The number of character positions NetSpool is to allocate in an output line for each double-byte character.

Value	Meaning
-------	---------

0	Allocate two print positions for each DBCS character. This is the default value set by NetSpool.
---	--

1	Allocate only one print position for each DBCS character.
---	---

S2NCNUM The column number of the next print position after the transparent data. If this field is zero, NetSpool determines the column number.

S2NLNUM The line number of the next print position after the transparent data. If this field is zero, NetSpool determines the line number.

S2WRKLEN The length of the data in the work area, S2WRKARA.

S2WRKARA The transparent-data control (TRN) and data to be added to the data stream. You can also add other SCS controls that NetSpool supports. Valid controls are described in “Appendix C. SCS Printer Simulation” on page 143.

Examples

1. To not add any transparent data to the beginning of the data set, set code 2 or 8 in register 15.
2. To add transparent data to the beginning of the data set, do the following:
 - Put the TRN control and transparent data in the S2WRKARA field.
 - Put the length of the TRN control and transparent data in S2WRKARA into the S2WRKLEN field.
 - Set code 1 or 9 in register 15.

Transparent Data Control Exit (APIPPTD2)

The Transparent Data Control exit (APIPPTD2) receives control whenever NetSpool encounters the Transparent Data control (TRN) in an SCS input data stream. Using this exit, you can inspect the transparent data and leave it unchanged, modify it, or delete it.

When you modify the transparent data, you can also add other SCS controls, in addition to the TRN control. The SCS controls supported by NetSpool are described in “Appendix C. SCS Printer Simulation” on page 143. The total number of bytes of modified data that you can include is 275. NetSpool does not call the Transparent Data Control exit (APIPPTD2) again for any transparent data included by this exit.

In order to format lines and pages, NetSpool keeps track of the next output print position, that is, the column number and line number at which the next character is to print. For transparent data, NetSpool increases the column and line number by the length of the transparent data, as specified in the TRN control.

The APIPPTD2 exit allows you to change the column and line number of the next print position *after* the transparent data. To assist in calculating the desired line and column numbers, NetSpool provides the exit with the column and line number of the next print position, *before* the transparent data. NetSpool also provides the exit with other page-formatting values.

Input to APIPPTD2

At input to the Transparent Data Control exit (APIPPTD2), the contents of the registers are:

Register 1	Address of the APIPP1X2 control block
Register 13	Address of an 18-word save area for saving the caller's registers
Register 14	Return address

The APIPP1X2 control block is described in Figure 21 on page 71. The following fields are valid at entry:

S2COL	The column number of the next print position
S2LM	Left margin column number
S2RM	Right margin column number
S2MPP	Maximum presentation position (line length)
S2LINE	The line number of the next print position
S2TM	Top margin line number (channel 1)
S2C2T12	Channel 2-12 line numbers
S2BM	Bottom margin line number
S2MPL	Maximum page length
S2PLUNAM	LU name of the application generating the print request (PLU)
S2LPTNAM	Name of the printer LU that NetSpool is processing (SLU)
S2PRTCLS	Logical-unit classes to which the printer LU belongs. Each bit in this field represents a class, with the high-order bit of the leftmost byte representing class 1. See Figure 21 on page 71 for a description of the bits.
S2TPDATA	Transparent data found in the input data stream, containing the TRN control and transparent data

Output from APIPTD2

At return from the Transparent Data Control exit (APIPTD2), restore the contents of registers 1 through 14, and place one of the following codes in register 15 to tell NetSpool what action to take:

Code Meaning

- | | |
|---|---|
| 1 | Modify the transparent data; S2WRKARA contains the modified data. |
| 2 | Do <i>not</i> modify the transparent data. |
| 4 | Close the VTAM session for this printer LU. |
| 8 | Do <i>not</i> call exit APIPTD2 again for any additional transparent data found in this data set. Do <i>not</i> modify the transparent data. |
| 9 | Do <i>not</i> call exit APIPTD2 again for any additional transparent data found in this data set. Modify the transparent data; S2WRKARA contains the modified data. |

Also, set the following fields in the APIPP1X2 control block.

- | | |
|-----------------|--|
| S2NCNUM | The column number of the next print position after the transparent data. If this field is zero, NetSpool determines the column number. |
| S2NLNUM | The line number of the next print position after the transparent data. If this field is zero, NetSpool determines the line number. |
| S2WRKLEN | The length of the data in the work area, S2WRKARA. Set this field to zero to delete the transparent data. |
| S2WRKARA | The modified TRN control and transparent data. You can also include other SCS controls that NetSpool supports. Valid controls are described in “Appendix C. SCS Printer Simulation” on page 143. |

Note: Do *not* change the TRN control or data in the S2TPDATA field.

Examples

1. To include the transparent data passed to this exit, unmodified, in the output data stream, set code 2 or 8 in register 15.
2. To modify the transparent data in the output data stream, do the following:
 - Put the TRN control and the modified transparent data in the S2WRKARA field.
 - Put the length of the TRN control and transparent data into S2WRKARA in the S2WRKLEN field.
 - Set code 1 or 9 in register 15.
3. To delete the transparent data passed to this exit from the output data stream, do the following:
 - Set the S2WRKLEN field to zero, to indicate that no data is in S2WRKARA.
 - Set code 1 or 9 in register 15.

Sample Exits and Macro

NetSpool provides sample exits and a macro, written in assembler language. The sample exits are in SYS1.SAMPLIB; the macro is in SYS1.MACLIB:

- APIJPTD1, a sample Beginning of File exit
- APIJPTD2, a sample Transparent Data Control exit
- APIPP1X2, a macro that contains parameter area S2STG and is used in assembling the exits.

You can either view the samples online or print them. You can then modify, assemble, and install the exit programs to provide your own versions of the exits. When you link-edit the samples, change the names to:

- APIPPTD1 for the Beginning of File exit and
- APIPPTD2 for the Transparent Data Control exit

Object code for the sample exits is not provided.

Installing the Exits

To install the exits, compile and link-edit them, as follows:

- Specify APIPPTD1 as the name of the Beginning of File exit. Specify APIPPTD2 the name of the Transparent Data Control exit.
- Specify the RENT link-edit attribute for improved performance; however, if you specify the RENT attribute, you must stop and restart NetSpool to pick up changes to the exit.
- Specify the AC(1) link-edit attribute and put the exit in an APF-authorized library.

You can identify the library containing the exits in either:

- A STEPLIB statement in the NetSpool startup procedure. To provide a different exit program for each instance of NetSpool, use a unique STEPLIB for each startup procedure.
- A library concatenated to LNKLIST. If you place an exit in a LNKLIST library, then you must use the same exit program for each instance of NetSpool.

Modifying the Exits

If you coded the exit to be reentrant and specified the RENT link-edit attribute, you must stop and restart NetSpool to pick up the new exit. If you did not specify the RENT link-edit attribute, you can replace the exit while NetSpool is running; however, IBM recommends that you code the exit to be reentrant for improved performance.

Chapter 8. Customizing IP PrintWay

The IP PrintWay component of Infoprint Server transmits output data sets from the JES spool to remote printers in a TCP/IP network using any one of these TCP/IP protocols: LPR to LPD, direct socket printing, or Internet Printing Protocol (IPP). See “IP PrintWay” on page 10 for an overview of IP PrintWay and how this component fits into your system.

This chapter describes the following required and optional tasks to customize IP PrintWay:

Task	Condition	See Page:
Customizing TCP/IP	Required	77 and 28
Customizing the IPP Client	Optional: To use IPP protocol	78
Defining the IP PrintWay Functional Subsystem	Required	79
APF-Authorizing SYS1.IMAGELIB	Optional: To format data using FCBs	86
Allocating and Initializing the Transmission-Queue Data Set	Required	86
Restricting Access to the Transmission-Queue Data Set	Optional: To limit access	87
Allocating and Initializing the IP PrintWay Message-Log Data Set	Required	87
Creating an IP PrintWay Startup Procedure	Required	88
Writing IP PrintWay Exit programs	Optional: To customize IP PrintWay processing	90
Customizing PSF for OS/390 for Sharing Network Printers	Optional: To share printers with PSF for OS/390	125

After customizing IP PrintWay, the following tasks must be performed; refer to *OS/390 Infoprint Server Operation and Administration* for information about these tasks.

- Create a printer definition for each target printer in the Printer Inventory and specify printer attributes required by IP PrintWay. If users can specify the IP addresses of the target printers on JCL, you do not need to create a printer definition for each printer.
- Create FSS and FSA definitions in the Printer Inventory for IP PrintWay unless default values are suitable.
- Start IP PrintWay.

Customizing TCP/IP

To use IP PrintWay you must customize TCP/IP as described in *OS/390 SecureWay Communications Server: IP Configuration*. When you customize TCP/IP, you can customize the TCPIP.DATA and PROFILE.TCPIP data sets.

Customizing TCPIP.DATA Data Set

The TCPIP.DATA data set contains the following statements used by IP PrintWay:

TCPIPJOBNAME

This statement specifies the name of the TCP/IP program in your installation. If you have several TCP/IP programs installed with different names, the administrator can specify the name of the TCP/IP program that IP PrintWay is to use in the FSS definition for the IP PrintWay FSS in the Printer Inventory; refer to *OS/390 Infoprint Server Operation and Administration* for information about FSS definitions. If you specify a TCP/IP name in both places, IP PrintWay uses the name in the FSS definition.

If you do not specify a name in either the TCPIPJOBNAME statement or in the FSS definition, the name of the TCP/IP program must be 'TCPIP', which is the TCP/IP default name.

DATASETPREFIX

The DATASETPREFIX statement specifies the high-level qualifier for TCP/IP data sets. IP PrintWay uses this high-level qualifier when searching for TCP/IP translation tables.

If you create a unique TCPIP.DATA data set for use by IP PrintWay, you can identify that data set in the IP PrintWay startup procedure, using a //SYSTCPD DD statement or by naming the data set using the jobname of the IP PrintWay startup procedure as the high-level qualifier.

To find the TCPIP.DATA data set, IP PrintWay follows this search sequence, using the first value found for each statement. If an allocation fails, the data set does not exist, or the data set is not available, IP PrintWay searches the next data set in the sequence.

1. The data set defined by the SYSTCPD DD statement in the IP PrintWay startup procedure.
2. A data set named *jobname*.TCPIP.DATA, where *jobname* is the jobname of the IP PrintWay startup procedure.
3. A data set named SYS1.TCPPARMS(TCPDATA)
4. A data set named *hlq*.TCPIP.DATA, where *hlq* is the TCP/IP high-level qualifier found earlier in the search sequence or the system default value of 'TCPIP'.

If you change any of the values in the TCPIP.DATA data set, restart TCP/IP to pick up the changes.

Customizing the PROFILE.TCPIP Data Set

When you customize TCP/IP, you can customize the PROFILE.TCPIP data set. See "Customizing TCP/IP" on page 28 for information about how to customize this data set for IP PrintWay and also for other components of Infoprint Server.

Customizing the IPP Client

If your installation uses the IPP protocol in IP PrintWay, the Java libraries **libjava.a**, **libjtc.so**, and **libzip.so** must be APF-authorized, that is, they must reside in an APF-authorized library.

If the JAVA_HOME environment variable names the Java directory, you can use the following OS/390 UNIX shell commands to APF-authorize these libraries:

```
cd $JAVA_HOME/lib/mvs/native_threads
extattr +a libjava.a libjtc.so libzip.so
```

You must be permitted to the BPX.FILEATTR.APF facility class in order to run the **extattr** command. A sample job is provided in SYS1.SAMPLIB(AOPJAUTH) to perform this function.

Note: When maintenance is applied to Java, APF authorization is lost; therefore, you must rerun AOPJAUTH after maintenance is applied to Java.

If your installation did not install Infoprint Server files in the default directories, also specify the ANFLIB and ANFCALL DD statements in the IP PrintWay startup procedure; see “Creating an IP PrintWay Startup Procedure” on page 88 for more information.

Defining the IP PrintWay Functional Subsystem

IP PrintWay operates as a JES functional subsystem (FSS). A functional subsystem is an extension of JES that runs in its own address space. For more information about JES functional subsystems, refer to one of the following publications:

- *OS/390 JES2 Initialization and Tuning Guide*
- *OS/390 JES3 Initialization and Tuning Guide*

Within the IP PrintWay FSS, you define one or more IP PrintWay functional subsystem applications (FSAs), which use the support facilities of the FSS to communicate with JES. Several IP PrintWay FSAs can run in the same FSS. The following sections describe planning decisions to consider before coding the JES2 or JES3 initialization statements and writing a startup procedure.

Determining How Many Functional Subsystems to Define

You do not need to create more than one IP PrintWay FSS; however, for improved throughput and more efficient use of system resources, you can define up to 2000 FSSs.

If you define more than one IP PrintWay FSS, specify the same transmission-queue data set in the startup procedure for each FSS to ensure that data sets are always transmitted in the same order as IP PrintWay acquires them from the JES spool. Note that when multiple IP PrintWay FSSs share a transmission-queue data set, you must also allocate the data set with SHAREOPTIONS (4 3). When you define the FSSs in this way, each FSS reads all of the transmission-queue entries and ensures that data sets are sent in the right order. See “Allocating and Initializing the Transmission-Queue Data Set” on page 86 for a sample job for allocating the transmission-queue data set.

Determining How Many Functional Subsystem Applications to Define

You do not need to create more than one IP PrintWay FSA because one PrintWay FSA can transmit data sets to many printers or print servers; however, for improved throughput and more efficient use of system resources, you can define up to 64 FSAs per FSS; however, IBM recommends that you do not create more than 35 FSAs per FSS.

As a guideline to the number of FSAs to use, you should determine the burst rate (that is, how many jobs must print concurrently) and specify at least that many FSAs. For example, some installations with low print volume can drive as many as 600 printers with 5 or 6 FSAs.

Determining Work-Selection Criteria

You define work-selection criteria for each IP PrintWay FSA during JES initialization. These criteria can determine which output data sets each FSA selects from the JES spool. You specify the work-selection criteria on the WS parameter of either the JES2 PRTnnnnn statement or the JES3 DEVICE statement. JES2 and JES3 let you specify a variety of work-selection criteria, which correspond to JCL parameters. Some work-selection criteria that you might consider for data sets to be processed by IP PrintWay are: output class, form name, and destination name. Some considerations related to work-selection criteria are:

- Do not specify destination as a work-selection criterion if your installation plans to specify the host name or IP address in the DEST:IP parameter on OUTPUT JCL statements. The reason for this is that JES does *not* use the host name or IP address specified as an OUTPUT JCL parameter when determining whether an output data set meets the work-selection criteria defined for an FSA.
- In a JES3 environment, specify form as a work-selection criterion if your installation wants IP PrintWay to use the form name on OUTPUT JCL statements to select any printer definitions in the Printer Inventory. The reason for this is that JES3 passes the form name specified on an OUTPUT JCL statement to the IP PrintWay FSA only if form is a JES work-selection criterion. Although you specify form as a work-selection criterion, the IP PrintWay FSA selects jobs with all form names for processing.
- In a JES3 environment, specify FCB as a work-selection criterion if you want IP PrintWay to perform FCB processing. The reason for this is that JES3 passes the FCB name specified as an OUTPUT JCL parameter to the PrintWay FSA only when FCB is a JES work-selection criterion.

Defining the IP PrintWay Functional Subsystem in JES2

To define a functional subsystem for IP PrintWay in a JES2 environment, code these JES2 initialization statements:

- One FSS(*fss_name*) statement to define each IP PrintWay functional subsystem (FSS)
- One PRTnnnnn statement for each functional subsystem application (FSA) under control of the FSS

In addition to the FSS(*fss_name*) and PRTnnnnn statements, these JES2 initialization statements have considerations for IP PrintWay:

- On the JES2 SPOOLDEF statement, specify TRKCELL=5 for improved performance. Also, specify TRKCELL=YES on the PRTnnnnn statement.
- On the JES2 OUTCLASS statement, specify BLNKTRNC=NO if you do not want JES2 to truncate trailing blanks for data sets in an output class that IP PrintWay processes. The setting of this parameter can affect the formatting of the printed output. The default is BLNKTRNC=YES.

Sample JES2 Initialization Statements

Figure 22 shows sample JES2 statements for one FSS named PRINTWAY and one associated FSA named PRT123.

```
FSS(PRINTWAY)  PROC=ANFWPROC,AUTOSTOP=YES
PRT123         CLASS=E,FSS=PRINTWAY,MODE=FSS,PRESELECT=YES,
               START=NO,TRKCELL=YES,WS=(Q)
```

Figure 22. Sample JES2 Initialization Statements. These statements are commonly found in the initialization-deck member of SYS1.PARMLIB.

The following section describes the JES2 initialization statements shown in Figure 22 on page 80, including some additional parameters that you might want to specify for an IP PrintWay FSS and FSA.

Note: These sample statements do not include all possible JES parameters. For a complete list of JES parameters and detailed descriptions of each of the parameters, refer to *OS/390 JES2 Initialization and Tuning Reference*.

FSS(fss_name) Statement

The FSS(*fss_name*) initialization statement is optional but recommended. If you omit it, JES2 generates a default FSS when an FSA is started.

Each FSS must have a unique name, specified as a one- to eight-character name. In the example, the FSS name is PRINTWAY. This FSS name must match the FSS name in the PRTnnnnn statement for each associated FSA.

FSS statement parameters are:

PROC=procedure_name

Specifies the name of the procedure for starting this FSS. Different FSS statements can refer to the same startup procedure. See “Creating an IP PrintWay Startup Procedure” on page 88 for more information. This parameter is required.

AUTOSTOP=YES | NO

Specifies whether or not the FSS address space is to be stopped automatically if all FSAs under control of the FSS are stopped. Specifying AUTOSTOP=YES can conserve system resources. This parameter is optional; if you omit it, the default is NO.

HASPFSSM=name | HASPFSSM

Specifies the 1- to 8-character name of the load module to be loaded into the functional subsystem address space. This load module contains the various JES2-supplied FSI service routines. If your installation uses the default libraries and module names for the JES2 component, as distributed by IBM, omit this parameter or code HASPFSSM=HASPFSSM.

PRTnnnnn Statement

A PRTnnnnn statement defines the FSA to JES. This section describes the PRTnnnnn parameters that you might want to specify for an IP PrintWay FSA. Refer to *OS/390 JES2 Initialization and Tuning Reference* for a complete list of parameters.

Each FSA must have a unique name; therefore, *nnnnn* must be a unique number from 1 to 32767.

If you create an FSA definition in the Printer Inventory for this FSA, specify this FSA name in the FSA definition. Although JES might let you use different formats to specify the FSA name in this statement, for example, PRT(*nnnnn*), specify the FSA name as PRTnnnnn in the FSA definition.

PRTnnnnn statement parameters include:

CLASS | QUEUE=(classes)

Specifies the output classes processed by this FSA. List all classes to be selected by this FSA; do not separate each class with a comma. If you designate CL or Q as a work-selection criterion on the WS parameter, the FSA processes data sets whose class matches one of the values specified here.

FORMS=(*form_names*)

Specifies the one- to eight-character form names processed by this FSA. List from one to eight different form names that can be selected by this FSA; separate each form name with a comma.

FSS=*fss_name*

Specifies the name of the FSS associated with this FSA. This parameter is required.

MODE=FSS

Specifies that the FSA is managed by an FSS. This parameter is optional. The default is FSS if you code the FSS parameter.

PRESELECT=YES | NO

Specifies whether output data sets are preselected for this FSA. This parameter is optional. If you omit it, the default is YES.

ROUTECD=(*destinations*)

Specifies the destinations processed by this FSA. List from one to four different destination names that can be selected by this FSA; separate each destination name with a comma.

START=YES | NO

Specifies whether or not JES2 is to automatically start this FSA whenever JES2 starts. If you specify START=NO, the operator must start the FSA. Consider specifying START=NO so that TCP/IP can be started before IP PrintWay. This parameter is optional. If you omit it, START=YES is the default.

TRKCELL=YES | NO

Specifies whether or not track-cell despooling is to be used with this FSA. You specify the size of the track cell, in terms of buffers, in the TRKCELL parameter of the JES2 SPOOLDEF statement. For improved performance, IBM recommends that you specify TRKCELL=YES on this statement and TRKCELL=5 on the JES2 SPOOLDEF statement. The default is NO.

WS=(*work_selection_criteria*)

Specifies the work-selection criteria for this FSA; separate each value with a comma. Refer to *OS/390 JES2 Initialization and Tuning Reference* for the valid values and defaults. For an IP PrintWay FSA, consider the following values:

- | | |
|---------------|--|
| CL Q | Specifies that the FSA selects only those data sets with the same class as specified in the CLASS parameter of this statement. |
| R | Specifies that the FSA selects only those data sets with the same destination as specified in the ROUTECDE parameter of this statement. For considerations related to this work-selection criterion, see “Determining Work-Selection Criteria” on page 80. |
| F | Specifies that the FSA selects only those data sets with the same form name as specified in the FORMS parameter of this statement. |

This parameter is optional but recommended. If you omit it, the FSA selects output data sets for processing according to default work-selection criteria.

Note: You might want to specify the SEP=NO parameter to prevent transmission of the JESNEWS data set.

Defining the IP PrintWay Functional Subsystem in JES3

To define a functional subsystem for IP PrintWay in a JES3 environment, code these JES3 initialization statements:

- One FSSDEF statement to define each IP PrintWay functional subsystem (FSS)
- One DEVICE statement for each functional subsystem application (FSA) under control of the FSS

In addition to specifying the FSSDEF and DEVICE statements, on the JES3 SYSOUT statement, specify TRUNC=NO if you do not want JES3 to truncate trailing blanks for data sets in an output class that IP PrintWay processes. The setting of this parameter can affect the formatting of the printed output. The default is the value set in the TRUNC parameter of the JES3 BUFFER statement or TRUNC=YES.

Sample JES3 Initialization Statements

Figure 23 shows sample JES3 statements for one FSS named PRINTWAY and one associated FSA named PRT1.

```
FSSDEF,FSSNAME=PRINTWAY,PNAME=ANFWPROC,TYPE=WTR
DEVICE,DTYPE=PRTAFP1,FSSNAME=PRINTWAY,JNAME=PRT1,JUNIT=(,SYS1,UR,ON),
      MODE=FSS,WC=(E),WS=(CL,F)
```

Figure 23. Sample JES3 Initialization Statements. These statements are commonly found in the initialization-deck member of SYS1.PARMLIB. Continuation characters are not shown.

The following section describes the JES3 initialization statements shown in Figure 23, including some additional parameters that you might want to specify for an IP PrintWay FSS and FSA.

Note: These sample statements do not show all possible JES parameters. For a complete list of parameters and for detailed descriptions of each of the parameters, refer to *OS/390 JES3 Initialization and Tuning Reference*.

FSSDEF Statement

The FSSDEF initialization statement is optional but recommended. If you omit it, JES3 generates a default FSS when an FSA is started.

FSSDEF statement parameters are:

FSSNAME=*fss_name*

Specifies the name of this FSS. Each FSS must have a unique one- to eight-character name. This FSS name must match the FSS name in the DEVICE statement for each associated FSA. This parameter is required.

MSGDEST (*dest_class* | **JES**)

Specifies the console destination class for messages concerning this FSS. This parameter is optional; if you omit it, JES sets a default.

PNAME=*procedure_name*

Specifies the name of the procedure for starting this FSS. Different FSSDEF statements can refer to the same startup procedure. See “Creating an IP PrintWay Startup Procedure” on page 88 for more information. This parameter is required.

SYSTEM=*system_name*

Specifies the JES3 processor on which the FSS will run. This parameter is optional; if you omit it, JES determines the default from the DEVICE statement.

TERM=YES | NO

YES specifies that the FSS will terminate if the JES3 global address space is terminated by a *RETURN or *DUMP operator command. This parameter is optional; if you omit it, the default is NO.

TYPE=WTR

Specifies that the FSS is an output writer. This parameter is required.

DEVICE Statement

A DEVICE statement defines each FSA to JES. This section describes DEVICE parameters that you might want to specify for an IP PrintWay FSA. Refer to *OS/390 JES3 Initialization and Tuning Reference* for a complete list of parameters.

DGROUP=*destination_name*

Specifies a one- to eight-character destination processed by this FSA. If you designate destination as a work-selection criterion on the WS parameter, this FSA selects data sets that match the value specified. This parameter is optional. Specify this parameter only if you specify destination as a work-selection criteria on the WS parameter.

DTYPE=PRTAFP1

Specify PRTAFP1 as the device type for an IP PrintWay FSA. This parameter is required.

DYNAMIC=YES | NO

Specifies whether JES3 is to start and stop this FSA dynamically.

YES Specifies that JES3 is to start this FSA whenever work is available for it. JES3 stops this FSA, and deactivates the address space when no work is available.

NO Specifies that the operator is to start and stop this FSA. Specify NO to keep the address space active between transmission of data sets.

This parameter is optional. If you omit it, the default is NO.

FSSNAME=*fss_name*

Specifies a unique FSS associated with this FSA. This value must match the value coded for the FSSNAME parameter in the corresponding FSSDEF statement. This parameter is optional. If you omit it, the default is the name of this FSA, as specified in the JNAME parameter.

JNAME=*fsa_name*

Specifies the unique one- to eight-character name of this FSA. This parameter is required.

JUNIT= (*devnum*,*main*,*msgdest*,ON | OFF)

devnum For IP PrintWay, do *not* specify a device address; use a comma as a position holder.

main Name of the processor on which IP PrintWay is running.

msgdest Destination for messages about the FSA.

ON | OFF Specifies whether the FSA is initially online or offline

This parameter is required.

MODE=FSS

Specifies that this FSA is managed by an FSS. This parameter is required.

PDEFAULT={NONE | CHARS | FCB | CHARS,FCB}

Specifies a subset (CHARS and FCB) of JES3 default values that JES3 does not supply to IP PrintWay.

CHARS	Indicates that JES3 will not supply default CHARS and UCS to IP PrintWay for output that does not specifically request a CHARS or UCS value.
FCB	Indicates that JES3 will not supply default FCB or carriage tape to IP PrintWay for output that does not specifically request an FCB or carriage tape value.
NONE	Indicates that JES3 will supply default CHARS, UCS, and FCB to IP PrintWay.

This parameter is optional. The default is NONE.

WC=(classes)

Specifies the output classes processed by this FSA. List all classes to be selected by this FSA; separate each class with a comma. If you designate CLASS as a work-selection criterion on the WS parameter, this FSA selects data sets that match the values specified here. This parameter is optional. If you omit it, this FSA selects data sets with any output class.

WS=(work_selection_criteria)

Specifies the work-selection criteria for this FSA; separate each value with a comma. Refer to *OS/390 JES3 Initialization and Tuning Reference* for the valid values and the default. For an IP PrintWay FSA, consider the following values.

C	Causes JES3 to pass the FCB name specified on the OUTPUT JCL statement to the FSA.
CL	Specifies that the FSA selects only those data sets with the same class as specified in the WC parameter of this statement.
D	Specifies that the FSA selects only those data sets with the same destination as in the DGROUP parameter of this statement. For considerations related to this work-selection criterion, see “Determining Work-Selection Criteria” on page 80.
F	Requests that JES3 pass the forms name specified on the DD or OUTPUT JCL statement to the FSA.

If you want IP PrintWay to use the form name specified on OUTPUT JCL statements to select any printer definitions in the Printer Inventory, specify WS=(F) as one of the work-selection criteria. If you do not specify forms as a work-selection criterion, JES3 does not pass the form name specified in JCL to the IP PrintWay FSA.

If you want PrintWay to use the FCB to format output, specify WS=(C) as one of the work-selection criteria. If you do not specify FCB as a work-selection criterion, JES3 does not pass the FCB name specified in JCL to the PrintWay FSA.

Note: You might want to specify the BURST=NO parameter to prevent transmission of the JESNEWS data set.

APF-Authorizing SYS1.IMAGELIB

If you plan to use FCB support in IP PrintWay, you must link-list and APF authorize the SYS1.IMAGELIB data set. To temporarily authorize this data set, use the SETPROG APF system command. To permanently authorize this data set:

- Update the PROGxx member of SYS1.PARMLIB to include the data sets in the LNKLIST statement.
- If the LNKAUTH parameter that is placed in the IEASYSxx member of SYS1.PARMLIB does not specify LNKLIST, also add the data sets to the APF statement in the PROGxx member.

As an alternative to updating the PROGxx member, you can update the LNKLISTxx and IEAAPFxx members of SYS1.PARMLIB; however, IBM recommends updating the PROGxx member. For more information, refer to *OS/390 MVS Initialization and Tuning Reference*.

Allocating and Initializing the Transmission-Queue Data Set

The transmission-queue data set is a required data set that contains an entry for each data set that IP PrintWay is processing. IP PrintWay creates the entries in the transmission-queue data set. You cannot add entries to it; however, you can use the Infoprint Server ISPF panels to browse, modify, and delete entries. For information about how to use the Infoprint Server ISPF panels to view the transmission-queue data set, refer to *OS/390 Infoprint Server Operation and Administration*.

You can allocate one transmission-queue data set to be shared by all IP PrintWay functional subsystems (FSSs). Or, you can create a separate data set for use by each IP PrintWay FSS. If more than one FSS can transmit output data sets to the same print queue, IBM recommends sharing the transmission-queue data set so that data sets are printed in the correct order.

The transmission-queue data set can be shared with other instances of IP PrintWay running at the *same* level on other systems; however, because the format of this data set has changed in OS/390 V2R8, it *cannot* be shared with another instance of IP PrintWay running at a *previous* level.

Note: When multiple FSSs share a transmission-queue data set, you must allocate the data set with SHAREOPTIONS (4 3).

The IP PrintWay transmission-queue data set is a VSAM key-sequenced data set. Use the IDCAMS program with the specifications shown in Figure 24 on page 87 to allocate it. Refer to *DFSMS/MVS Access Method Services for ICF* if you need more information about allocating VSAM data sets.

You can define the IP PrintWay transmission-queue data set with any name. You might want to use the IP PrintWay default data set name **ANF.QUEUE**. Unless you define another default, this default appears on the IP PrintWay ISPF panels. This default is also used in the sample JCL provided by IP PrintWay.

Use the sample JCL in SYS1.SAMPLIB(ANFDEAL) to allocate the IP PrintWay transmission-queue data set. You can copy and modify this JCL for your installation.

```

DEFINE CLUSTER
  (NAME(ANF.QUEUE) -
   VOLUMES(volser) -
   INDEXED -
   SPEED -
   SHAREOPTIONS(4 3)) -
  DATA -
  (NAME(ANF.QUEUE.DATA) -
   TRACKS(30 5) -
   KEYS(20 0) -
   RECORDSIZE(238 2048) -
   FREESPACE(10,10) -
   CISZ(4096)) -
  INDEX -
  (NAME(ANF.QUEUE.INDEX))

```

Figure 24. IDCAMS Parameters to Allocate a Transmission-Queue Data Set

Notes:

1. Specify the correct data set name for your installation in the NAME parameter. Replace *volser* with the volume ID.
2. Use the values for the RECORDSIZE, SHAREOPTIONS, and KEYS options shown in Figure 24.
3. Adjust the value of the FREESPACE option based on experience with the data set. The size of the PrintWay transmission-queue data set changes dynamically, according to the number of data sets retained after successful or unsuccessful transmission.

After allocating space for the transmission-queue data set, use the sample JCL in SYS1.SAMPLIB(ANFQINIT) to initialize the data set with binary zeroes. If you did not use the default name, ANF.QUEUE, for the data set, change the name of the data set in the sample JCL.

Restricting Access to the Transmission-Queue Data Set

All ISPF users can use the Infoprint Server ISPF panels to update the IP PrintWay transmission-queue data set, even if you have established a RACF administration group for the Printer Inventory. You can, however, use Resource Access Control Facility (RACF) or a similar program to restrict access to the data set.

If you have RACF protected the transmission-queue data set, the Infoprint Server ISPF panels display only the actions that a user is allowed to perform. For example, if a user has read-only access to the transmission-queue data set, that user is allowed only to browse entries in the data set and is *not* allowed to modify or delete entries.

Allocating and Initializing the IP PrintWay Message-Log Data Set

IP PrintWay writes messages that track data-set transmissions in an IP PrintWay message-log data set. This is a required data set. The message log is organized as a circular queue. When a message extends to the end of the data set, the next message starts at the beginning. In this way, the size of the message data set remains stable, with old messages automatically overlaid by newer messages.

You can create one message-log data set to be shared by all IP PrintWay functional subsystems (FSSs); or you can create separate data sets for each FSS. To allocate space for the data set and initialize it, use the sample JCL in SYS1.SAMPLIB(ANFMIJCL) to run the ANFMFILE program.

The message-log data set requires the following specifications:

- Sequential organization (PS)
- Fixed block format (FBA)
- Logical record length (LRECL) of 57 at a minimum; 80 is recommended
- Block size (BLKSIZE) of about 10 records per block; 800 is recommended

Secondary space is unnecessary, because the ANFMFILE program formats all available space. To change the size of the IP PrintWay message-log data set, rerun the ANFMFILE program.

Note: The ANFMFILE program produces a D37 abend when it completes processing.

You can suppress selected messages that IP PrintWay writes to the message-log data set by writing a Message exit. See “Message Exit (ANFUXMSG)” on page 117 for more information.

You can view messages in the message-log data set by using the Infoprint Server ISPF panels. Refer to *OS/390 Infoprint Server Operation and Administration* for information.

Creating an IP PrintWay Startup Procedure

Before starting an IP PrintWay FSS, you must create a cataloged startup procedure. The same startup procedure can be used to start more than one IP PrintWay FSS if each FSS shares the same transmission-queue data set. You specify the name of the startup procedure in the JES initialization statement for each FSS.

The startup procedure identifies:

- The program name, region size, and time-out value
- Pathname of the Printer Inventory
- Location of the IP PrintWay transmission-queue and message-log data sets
- Location of IPP and Java libraries required by the IP PrintWay IPP client

Include the startup procedure in a procedure library known to JES. For more information on procedure libraries, refer to *OS/390 JES2 Initialization and Tuning Guide* and *OS/390 JES3 Initialization and Tuning Guide*.

Note: If you change the IP PrintWay startup procedure, you must stop all IP PrintWay FSAs within the FSS that uses this startup procedure and then restart the FSAs to pick up the changes. If JES2 does not stop the FSS automatically, you also need to stop the FSS before restarting the FSAs. Refer to *OS/390 Infoprint Server Operation and Administration* for information about how to stop and start IP PrintWay.

Startup Procedure

IP PrintWay provides a startup procedure in SYS1.PROCLIB(ANFWPROC), which is shown in Figure 25 on page 89. You can copy and modify this sample procedure to suit your installation. JES2 initialization statements for an FSS that uses this startup procedure are shown in Figure 22 on page 80. JES3 initialization statements are shown in Figure 23 on page 83.

```

//ANFWPROC PROC HLQ=ANF,
//              TCPHLQ=TCPIP
//* -----
//* this is a sample procedure for starting IP PrintWay
//* -----
//IEFPROC EXEC PGM=ANFFIEP,REGION=1M,TIME=NOLIMIT,PARM='INV=AOP1'
//*-----
//* AOP1 is the default name of the printer inventory
//* server. This must be modified if this is not the
//* name being used for the inventory server
//*-----
//ANFQUEUE DD DSN=&HLQ..QUEUE,DISP=SHR
//SYSTCPD DD DSN=&TCPHLQ..TCPIP.DATA,DISP=SHR
//ANFMSG DD DSN=&HLQ..MSGFILE,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSABEND DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*

```

Figure 25. IP PrintWay Startup Procedure — SYS1.PROCLIB(ANFWPROC)

Notes:

1. If you have written an installation exit, add a STEPLIB statement to identify the library containing the exit, unless the library is part of the standard OS/390 search order.
2. If you did not add the Language Environment run-time library, CEE.SCEERUN, to the system LNKLIST, add the SCEERUN data set to the STEPLIB DD statement.
3. If you plan to use the IP PrintWay IPP client and have not added the C++ run-time library, CBC.SCLBDLL, to the system LNKLIST, add the SCLBDLL data set to the STEPLIB DD statement.
4. If you plan to use the IP PrintWay IPP client and have not installed Infoprint Server files in the default libraries, add the ANFLIB and ANFCALL DD statements, as shown in “JCL Statements”.

JCL Statements

This section describes the statements and parameters shown in Figure 25.

//proc_name PROC HLQ=ANF, TCPHLQ=TCPIP

Names a cataloged procedure. *proc_name* must match the procedure name in a JES initialization statement for this FSS:

- PROC parameter of the JES2 FSS(*fss_name*) statement
- PNAME parameter of the JES3 FSSDEF statement

The HLQ keyword identifies the high-level qualifier for the IP PrintWay data sets; TCPHLQ identifies the high-level qualifier for the TCPIP.DATA data set. Change these values for your installation if necessary.

//label EXEC

PGM=ANFFIEP,REGION=1M,TIME=NOLIMIT,PARM='INV=*inventory*'

Invokes the IP PrintWay program.

PGM=ANFFIEP

The IP PrintWay program entry point.

REGION=1M

The amount of below-the-line storage required by the FSS. Specify 1M, regardless of the number of FSAs supported by the FSS.

TIME=NOLIMIT

Specify TIME=NOLIMIT to prevent TIMEOUT abends.

PARM='INV=*inventory*'

The name that you assigned to the Printer Inventory in the **inventory** statement in the **aopd.conf** configuration file. Specify AOP1 (the default name) if you have not created an **aopd.conf** configuration file or if the **inventory** statement is omitted. This name is case sensitive. This parameter is required.

//ANFQUEUE DD DSN=&HLQ..QUEUE,DISP=SHR

The IP PrintWay transmission-queue data set. This statement is required.

Note: To ensure that data sets are always transmitted in the same order as IP PrintWay acquires them from the JES spool, specify the same transmission-queue data set for each IP PrintWay FSS.

//SYSTCPD DD DSN=&TCPHLQ..TCPIP.DATA,DISP=SHR

The TCPIP.DATA data set. This statement is optional. For a description of how IP PrintWay searches for the TCPIP.DATA data set, see “Customizing TCP/IP” on page 77.

//ANFMMSG DD DSN=&HLQ..MSGFILE,DISP=SHR

The IP PrintWay message-log data set. This statement is required.

//ANFLIB DD PATH='pathname'

The HFS directory that contains Infoprint Server dynamic link library (DLL) files. This statement is required only if your installation did not install Infoprint Server files in the default directory **/usr/lpp/Printsrv/lib/**.

//ANFCALL DD PATH='pathname'

The HFS directory that contains Infoprint Server call libraries. This statement is required only if your installation did not install Infoprint Server files in the default directory, **/usr/lpp/Printsrv/classes/**.

Writing IP PrintWay Exit programs

Note

This section contains programming-interface information.

The following optional exits allow you to customize IP PrintWay processing:

- Routing exit

The Routing exit allows you to inspect routing parameters for each data set before the data set is transmitted. You can change the routing parameters for the data set, hold the data set on the JES spool, or delete the data set from the JES spool. See “Routing Exit (ANFUXRTG)” on page 96 for more information.

- Begin Data Set exit

The Begin Data Set exit allows you to inspect the transmission options for a data set before the data set is transmitted. You can change transmission options and also add one or more records to the beginning of the data set. See “Begin Data Set Exit” on page 102 for more information.

- Record exit

The Record exit allows you to add, replace, or delete records in a data set before the data set is transmitted. See “Record Exit” on page 106 for more information.

- **End Data Set exit**

The End Data Set exit allows you to add records to the end of a data set before the data set is transmitted. See “End Data Set Exit” on page 111 for more information.

- **SMF exit**

The SMF exit allows you to replace or suppress the System Management Facilities (SMF) type-6 record that IP PrintWay writes for each data set that has been transmitted. See “SMF Exit (ANFUXSMF)” on page 115 for more information.

- **Message exit**

The Message exit allows you to suppress or change the text of any message that IP PrintWay writes to the IP PrintWay message-log data set. See “Message Exit (ANFUXMSG)” on page 117 for more information.

- **Response Notification Exit**

The Response Notification exit allows you to perform an action based on the success or failure of the transmission as indicated by the response notification code. See “Response Notification Exit (ANFUXRSP)” on page 119 for more information.

Each exit, except for the Message exit, can also create a message for IP PrintWay to issue to the operator’s console or to the IP PrintWay message-log data set. This allows the exit to inform the operator or system programmer about actions taken by the exit.

IP PrintWay calls the exits in the following order:

1. **Initial data set processing exits**

- **Routing Exit**

Once with each data set acquired from the JES spool

- **Response Notification Exit**

Once with each data set acquisition, if an error occurs that keeps IP PrintWay from transmitting the data set

2. **Data set transmission processing exits**

Data set transmission processing exits run in sequence. IP PrintWay does not invoke any other exits until after the data set has been transmitted to the remote system.

- **Begin Data Set Exit**

Once with each transmission attempt.

- **Record Exit**

Once with each record, after the Begin Data Set Exit completes.

- **End Data Set Exit**

Once with each transmission attempt, after the last Record Exit completes.

- **Response Notification Exit**

Once with each transmission attempt, after the End Data Set Exit completes.

3. **SMF Exit**

When a data set is to be purged from the JES queue, following the IP PrintWay retention rules.

IP PrintWay calls the Message exit before writing any message to the IP PrintWay message-log data set.

Programming Considerations

Consider the following points when coding an exit:

- You should be an experienced programmer who is aware of overall system implications.
- Do not issue an explicit or implied WAIT in the exit program, because it puts the IP PrintWay FSA into a WAIT state.
- The exits run in 31-bit addressing mode.
- Code references to fields in control blocks using the names associated with each field in the DSECT control block. Do not use absolute displacements, because these displacements can change from one release to another.
- Code the exits to be reentrant.
- All exit programs run in supervisor state with a protection key of 1.
- Programming exceptions cause IP PrintWay to abend; an ESTAI is in effect while the exit programs are running.
- Each IP PrintWay FSA calls all exits except for the Message exit serially, passing the same Exit Parameter Area (ANFUEXTP) to each exit. IP PrintWay passes the Message exit a separate copy of the ANFUEXTP control block.
- All exits, except for the Message exit, can obtain two work areas and pass pointers to these work areas in the Exit Parameter Area. Using the work areas, one exit can pass data to another exit or to the next invocation of the same exit.
- The data set transmission processing exits are invoked in sequence for the transmission of a single spool data set. This allows information specific to a spool data set to be passed from one of these exits to another, using the work area. Because the Routing exit, the SMF exit, and the Message exit are not invoked in sequence for a particular spool data set, you should not attempt to pass specific spool data set information using these exits.
- A return code of 8 to the Response Notification exit indicates that an error was encountered in IP PrintWay's initial processing of a data set. In this case, no data set transmission processing exits are invoked; therefore, no communication with these exits is possible.

Note: Future IP PrintWay program maintenance may require that you reassemble the exits.

Common Control Blocks

All exits, except for the Message exit, have access to data set and job information contained in the following control blocks:

- JES job separator page data area (IAZJSPA)
- Scheduler work block text unit (SWBTU)

All exits have access to data set and job information contained in the following control block:

- IP PrintWay Exit Parameter Area (ANFUEXTP)

JES Job Separator Page Data Area (IAZJSPA)

The JES job separator page data area (IAZJSPA) is a system control block that contains job and data set information. Refer to *OS/390 MVS Using the Functional Subsystem Interface* for more information on the contents of the IAZJSPA.

Scheduler Work Block Text Unit (SWBTU)

The SWBTU is a system control block that contains JCL parameters for the data set. These JCL values are merged from the DD statement, the OUTPUT statement, and JES defaults, according to a hierarchy determined by JES. You can use the SWBTUREQ macro to obtain the JCL parameters specified for the data set from the SWBTU. For example, you could obtain distribution parameters specified on the OUTPUT statement and print them on a separator page created in your Begin Data Set exit. Refer to *OS/390 Infoprint Server User's Guide* for information on how to specify distribution parameters on an OUTPUT statement.

For more information about the SWBTUREQ macro, refer to *OS/390 MVS Programming: Authorized Assembler Services Reference SET-WTO*. For more information about SWBs and how to use the SWB token to retrieve keywords, refer to *OS/390 MVS Using the Functional Subsystem Interface*.

IP PrintWay Exit Parameter Area (ANFUEXTP)

The IP PrintWay Exit Parameter Area (ANFUEXTP) provides information to the exits about the data set being processed. Also, the exits can set flags in ANFUEXTP to specify the action IP PrintWay is to take at return and can modify values in ANFUEXTP to change the processing of IP PrintWay.

The same ANFUEXTP control block is shared by all exits, except for the Message exit, that are called by the same IP PrintWay FSA; however, an exit program does not need to enqueue on this control block, because IP PrintWay calls the exits serially.

See the description of each exit for the fields that are valid at input and the fields that each exit can modify. Figure 26 on page 94 shows the format of ANFUEXTP. Macro ANFUEXTP provides a DSECT in assembler language for this control block.

Offsets	Hex	Type	Length	Name	Description
0	(0)	Structure	1280	ANFUEXT	Exit parameter area
0	(0)	Character	4	XTPID	Eye catcher - 'XTP '
4	(4)	Unsigned	4	XTPSAVE(18)	Save area for exit
76	(4C)	Address	4	XTPWORK1	Work area for exit
80	(50)	Address	4	XTPWORK2	Work area for exit
84	(54)	Character	53	XTPDSNAM	Data set name
137	(89)	Character	3	*	Reserved
140	(8C)	Character	8	XTPUSRID	User id of submitter
148	(94)	Bit string	1	XTPSRCFL	Flags showing where values came from. If the bit is on the value came from JCL on OUTPUT; otherwise it came from the printer definition or defaults:
		1... ..		XTPSOPTS	PRTOPTNS from JCL
		.1.. ..		XTPSIPAD	IPADDR from JCL
		..1.		XTPSPRTQ	PRTQUEUE from JCL
		...1		XTPSRTNS	RETAINS from JCL
	 1...		XTPSRTNF	RETAINF from JCL
	1..		XTPSRTYL	RETRYL from JCL
	1.		XTPSRTYT	RETRYT from JCL
	1		XTPSPORT	PORTNO from JCL
149	(95)	Bit string	1	XTPDSFLG	Data set information flags:
		1... ..		XTPDSCC	data set has carriage controls
		.1..		XTPDSHLD	release data set to JES; data set is held on JES spool
		..1.		XTPDSTRM	release data set to JES; data set is deleted from JES spool
		...1		XTPDSRTG	reselect printer definition
	 1...		XTPDSJOB	data set is first in job
	1..		XTPDSJNW	data set is JESNEWS data set
	1.		XTPDSEJB	data set is last in job
	1		XTPDSCON	data set is in a concatenation
150	(96)	Bit string	1	XTPRCFLG	Record flags:
		1... ..		XTPRCTRN	translate record from exit
		.1..		XTPRCCC	record from exit has carriage controls
		..1.		XTPRCORG	transmit original record
		...1		XTPRDFST	1st call to exit for record or data set
	 1...		XTPRLAST	last record from exit
	1..		XTPRCEXT	transmit record from exit
	1.		XTPRCDBL	record from exit is double-byte
	1		*	reserved
151	(97)	Bit string	1	XTPMSGFL	Message flags:
		1... ..		XTPM2CON	issue message to console
		.1..		XTPM2MDS	issue message to message data set
		..11 1111		*	reserved
152	(98)	Address	4	XTPERPTR	Address of record from exit
156	(9C)	Unsigned	4	XTPERLEN	Length of record from exit
160	(A0)	Address	4	XTPPRPTR	Address of original record
164	(A4)	Unsigned	4	XTPPRLEN	Length of original record
168	(A8)	Character	16	XTPOPTNM	Name of components
184	(B8)	Unsigned	2	XTPLOLEN	Length of LPR options
186	(BA)	Unsigned	2	XTPPQLEN	Length of name of print queue
188	(BC)	Unsigned	2	XTPIALEN	Length of IP address or host name

Figure 26. Format of the IP PrintWay Exit Parameter Area (ANFUEXT) (Part 1 of 2)

190	(BE)	Unsigned	2	XTPMSGLN	Length of message
192	(C0)	Character	256	XTPPRTQU	Name of print queue
448	(1C0)	Character	256	XTPIPADR	IP address/host name or URL
704	(2C0)	Unsigned	4	XTPRETNS	Success retain time (see note)
708	(2C4)	Unsigned	4	XTPRETNF	Failure retain time (see note)
712	(2C8)	Unsigned	4	XTPRTRYL	Retry limit
716	(2CC)	Unsigned	4	XTPRTRYT	Retry time (see note)
720	(2D0)	Character	17	XTPKEY	Name of printer definition
720	(2D0)	Character	8	XTPDEST	DEST
728	(2D8)	Character	1	XTPCLASS	CLASS
729	(2D9)	Character	8	XTPFORMS	FORMS
737	(2E1)	Bit string	1	XTPRECFM	Record format
738	(2E2)	Unsigned	2	XTPMRECL	Maximum record length
740	(2E4)	Address	4	XTPSWBTP	Address of SWBTU area
744	(2E8)	Unsigned	4	XTPSWBTL	Length of SWBTU area
748	(2EC)	Address	4	XTPJSPAP	Address of JSPA
752	(2F0)	Unsigned	4	XTPCOPYS	Number of copies
756	(2F4)	Address	4	XTPMSGP	Address of message
760	(2F8)	Unsigned	4	XTPIPORT	PORTNO value
764	(2FC)	Bit string	1	XTPCFLGS	Reserved
765	(2FD)	Character	3	*	Reserved
=====					
The following fields are used by the Block Letter program.					
=====					
768	(300)	Bit string	1	XTPBIFLG	Input flags for block program:
		1...		XTPBSLNT	slant the letters
		.1..		XTPBPRFM	performance (narrow) letters
		..1.		XTPBLJST	left justify letters
		...1		XTPBFRST	first call to block program
					for this string
	 1111		*	reserved
769	(301)	Bit string	1	XTPBOFLG	Output flags for block program:
		1...		XTPBLAST	this is the last line
		.111 1111		*	reserved
770	(302)	Unsigned	1	XTPBLENG	Length of output record
771	(303)	Character	1	*	Reserved
772	(304)	Character	8	XTPBSTRG	String to convert, blank padded
780	(30C)	Character	150	XTPBOUTP	Output string
930	(3A2)	Character	16	XTPBWRKA	Work area for block program
946	(3B2)	Character	2	*	Reserved
948	(3B4)	Signed	4	XTPRSPCD	Response code
=====					
The following fields are used by all exits.					
=====					
952	(3B8)	Unsigned	4	XTP_ADDR_TYPE	Value in XTPIPADR field:
					0 = IP address or host name
					1 = URL of printer
956	(3BC)	Unsigned	4	XTP_ROUTING_KEY_TYPE	Value in XTPKEY field:
					0 = DEST, CLASS, and FORMS
					1 = Printer definition name
960	(3C0)	Character	40	*	Reserved
1000	(3E8)	Character	868	XTPOPTNS	Transmission options
					(See macro for field names)

Figure 26. Format of the IP PrintWay Exit Parameter Area (ANFUEXTP) (Part 2 of 2)

Note: The time in the XTPRETNS, XTPRETNF, and XTPRTRYT fields is a hexadecimal number, calculated as follows: $(\text{hours} \times 10\,000) + (\text{minutes} \times 100) + \text{seconds}$. For example, if the time period is 24 hours, 5 minutes, and 1 second, this field contains 03AB75, the hexadecimal representation of the decimal number 240501.

Routing Exit (ANFUXRTG)

The Routing exit allows you to inspect routing parameters for each data set before the data set is transmitted. When the Routing exit receives control, the routing parameters from the OUTPUT JCL statement for the data set and from the printer definition in the Printer Inventory have been merged. The parameters on the OUTPUT JCL statement always override the values specified in the printer definition. The XTPSRCFL field in the ANFUEXTP control block indicates whether or not each routing parameter was specified on the OUTPUT JCL statement.

If you determine in the exit that one or more routing parameters need to be changed, you can either (1) modify the routing parameters directly in the exit or (2) request that IP PrintWay obtain new routing parameters from a specific printer definition. You identify the new printer definition by the class, destination, and form name assigned to it. IP PrintWay does *not* call the Routing exit again for this data set after obtaining parameters from the new printer definition.

Instead of changing routing parameters, you can request that IP PrintWay have JES hold the data set on the JES spool for operator intervention, or you can request that IP PrintWay delete the data set from the JES spool.

IP PrintWay does *not* issue any messages to the IP PrintWay message log or to the job submitter when an exit requests that a data set be held or deleted. However, you can request that IP PrintWay issue a message created by the exit either to the operator's console or to the IP PrintWay message-log data set. IP PrintWay assigns the same message ID to each message created by an exit: ANFM155I.

IP PrintWay calls the same Routing exit before processing each data set. You can install different Routing exits for different IP PrintWay FSSs. See "Installing Exits" on page 124 for information.

You might want to write a Routing exit to perform one of the following functions:

- Enforce limits in JCL parameters. For example, your installation might want to limit the length of the retention time specified by job submitters on the OUTPUT JCL statement. To do this, your Routing exit could inspect the values specified in the RETAINS and RETAINF JCL parameters and change them if they exceed the maximum time defined for your installation.
- Restrict the use of certain printers to specific users. To do this, your Routing exit could inspect (1) the user ID of the job submitter and (2) the host name or IP address and the print queue name or port of the target printer. Your exit could hold or delete a data set if the user ID of the job submitter is not on your list of authorized users IDs.

Note: IP PrintWay does *not* call this exit if it cannot find a printer definition for a data set and the DEST=IP parameter is not specified on the OUTPUT JCL statement. Instead, IP PrintWay requests that JES hold the data set for operator intervention.

Input to Routing Exit

At entry to the Routing exit, the registers contain the following:

- | | |
|--------------------|--|
| Register 1 | Pointer to the address of the ANFUEXTP control block. The ANFUEXTP control block contains addresses of the IAZJSPA and the SWBTU control blocks. |
| Register 13 | Address of an 18-word save area for saving the caller's registers. |
| Register 14 | Return address. |

Register 15 Entry-point address.

The following fields in the ANFUEXTP control block are valid when the Routing exit is called. The format of ANFUEXTP is shown in Figure 26 on page 94.

XTPSAVE A register save area for use by the exit.

XTPWORK1 Address of a work area previously obtained by an exit. This field contains zeroes if a work area has not been obtained.

XTPWORK2 The address of a second work area previously obtained by an exit. This field contains zeroes if a work area has not been obtained.

XTPDSNAM Fully-qualified name of the data set being processed, padded to the right with blanks.

XTPUSRID User ID of the job submitter, padded to the right with blanks.

XTPSRCFL Flags indicating whether or not a parameter was specified on the OUTPUT JCL statement for the data set.

If the flag is set to 1, the indicated parameter was specified on the OUTPUT JCL statement.

If the flag is set to 0, the JCL parameter was not specified. Instead, IP PrintWay obtained the value from a printer definition. If the DEST=IP parameter was specified in JCL without the name of a printer definition, IP PrintWay obtained the value from the IP PrintWay default printer definition; if no PrintWay default printer definition exists, IP PrintWay sets a default value.

Flag	Meaning When Set to B'1'.
------	---------------------------

XTPSOPTS	The PRTOPTNS parameter specified the name of components in the Printer Inventory.
XTPSIPAD	The DEST=IP parameter specified the host name or IP address of the target system.
XTPSPRTQ	The PRTQUEUE parameter specified the name of the target print queue.
XTPSRTNS	The RETAINS parameter specified the retention time after a successful transmission.
XTPSRTNF	The RETAINF parameter specified the retention time after a failed transmission.
XTPSRTYL	The RETRYL parameter specified the retry limit.
XTPSRTYT	The RETRYT parameter specified the retry time.
XTPSPORT	The PORTNO parameter specified the port number.

XTPDSFLG Flags containing data set information:

Flag	Meaning When Set to B'1'.
------	---------------------------

XTPDSJOB	This is the first data set in the job.
XTPDSJNW	This data set is the JESNEWS data set.
XTPDSEJB	This is the last data set in the job.
XTPDSCON	This data set is part of a concatenation.

XTPOPTNM Name of components in the Printer Inventory that contain formatting and transmission options, as specified in the

	PRTOPTNS JCL parameter. The name is case sensitive and is padded to the right with blanks. This field contains blanks if no component name was specified in the PRTOPTNS parameter.
XTPPQLEN	Length of the print queue name in field XTPPRTQU.
XTPIALEN	Length of the value in field XTPIPADR.
XTPPRTQU	Name of the target print queue, as specified in the PRTQUEUE JCL parameter or in a printer definition.
XTPIPADR	Host name or IP address of the target system, as specified in the DEST=IP JCL parameter, in a printer definition; or the URL of the target system specified in the printer definition or the Routing exit. Field XTP_ADDR_TYPE identifies which value is in this field.
XTPRETNS	Period of time IP PrintWay is to retain the data set on the JES spool after a successful transmission, as specified in the RETAINS JCL parameter or in a printer definition. IP PrintWay may have set a default value of zeroes. See the note on page 95 for the format of this field.
XTPRETNF	Period of time IP PrintWay is to retain the data set on the JES spool after all transmission retries have failed, as specified in the RETAINF JCL parameter or in a printer definition. IP PrintWay may have set a default value of zeroes. See the note on page 95 for the format of this field.
XTPRTRYL	Number of times IP PrintWay is to retry transmission of the data set, as specified in the RETRYL JCL parameter or in a printer definition. IP PrintWay may have set a default value of zero.
XTPRTRYT	Period of time IP PrintWay is to wait between retries of the transmission, as specified in the RETRYT JCL parameter or in a printer definition. IP PrintWay may have set a default value of zero. See the note on page 95 for the format of this field.
XTPKEY	If XTP_ROUTING_KEY_TYPE is set to '1', contains the name of the printer definition, as specified in JCL or in the printer definition; the name is padded to the right with blanks. This field is case sensitive.
XTPDEST	If XTP_ROUTING_KEY_TYPE is set to '0', contains the destination name of the data set, as specified in JCL or defaulted by JES; the name is padded to the right with blanks. This field contains DFLTENTRY if the DEST=IP parameter was specified in the OUTPUT JCL statement without the name of a printer definition. This field contains blanks if the DEST field in the printer definition is not specified.
XTPCLASS	If XTP_ROUTING_KEY_TYPE is set to '0', contains the class of the data set, as specified in JCL or defaulted by JES. This field contains blanks if the CLASS field in the printer definition is not specified.
XTPFORMS	If XTP_ROUTING_KEY_TYPE is set to '0', contains the form name of the data set, as specified in JCL or defaulted by JES; the form name is padded to the right with blanks. This field contains blanks if the FORMS field in the printer definition is not specified.
XTPRECFM	Record format as indicated in the job file control block (JFCB). Refer to <i>OS/390 MVS Data Areas, Vol 3 (IVT-RCWK)</i> for a description of the JFCRECFM field in the JFCB.

XTPMRECL	Maximum record length
XTPSWBTP	Address of the SWBTU for the data set. See “Scheduler Work Block Text Unit (SWBTU)” on page 93 for more information.
XTPSWBTL	Length of the SWBTU.
XTPJSPAP	Address of the JES job separator page data area for the data set being processed (IAZJSPA). See “JES Job Separator Page Data Area (IAZJSPA)” on page 92 for more information.
XTPCOPYS	Number of copies of the data set, as specified in JCL or defaulted by JES.
XTPIPORT	Port number on the target system, as specified in the PORTNO JCL parameter or in a printer definition.
XTP_ADDR_TYPE	Value in field XTPIPADR: 0 = IP address or host name; 1 = URL
XTP_ROUTING_KEY_TYPE	Value in field XTPKEY: 0 = DEST, CLASS, and FORMS; 1 = printer definition name

Output from Routing Exit

At return, restore the contents of all registers. Do not set a return code in register 15.

The exit can modify the following fields in the ANFUEXTP control block:

XTPWORK1	Address of a work area obtained by this exit. This address is passed to every exit except the Message exit.
XTPWORK2	Address of a second work area obtained by this exit. This address is passed to every exit except the Message exit.
XTPDSFLG	Flags indicating the action you want IP PrintWay to take upon return. IP PrintWay sets these flags to B'0' before calling the exit. To transmit the data set, do <i>not</i> set any of these flags.

Note: Set only one of the following flags.

Flag	Meaning When Set to B'1'.
XTPDSHLD	Hold the data set on the JES spool for operator intervention.
XTPDSTRM	Delete the data set from the JES spool.
XTPDSRTG	Reselect the printer definition using the value in the XTPKEY field.
XTPMSGFL	Flags indicating the destination of a message created by this exit. IP PrintWay sets these flags to B'0' before calling the exit. Set both flags to B'1' to issue the message to both destinations. Place the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.
Flag	Meaning When Set to B'1'.
XTPM2CON	Issue the message to the operator's console.

XTPM2MDS Issue the message to the IP PrintWay message-log data set.

XTPMSGLN Length of the message whose address is in field XTPMSGP.

XTPMSGP Address of the message created by this exit. IP PrintWay adds a message ID, ANFM155I, to the beginning of this message.

If you set the XTPDSRTG flag, you can also change the following fields in the ANFUEXTP control block. (IP PrintWay ignores changes you make to fields containing routing parameters, such as fields XTPOPTNM, XTPPRTQU, XTPIPADR, XTPRETNS, XTPRETNE, XTPRTRYL, and XTPRTRYT.)

XTPKEY Name of the printer definition you want IP PrintWay to use. This field is case sensitive. Also set XTP_ROUTING_KEY_TYPE to '1'.

XTPCLASS Class assigned to the printer definition you want IP PrintWay to use. Do not use lowercase characters. Also set XTP_ROUTING_KEY_TYPE to '0'.

XTPDEST Destination name assigned to the printer definition you want IP PrintWay to use, padded to the right with blanks. Do not use lowercase characters. Also set XTP_ROUTING_KEY_TYPE to '0'.

Note: Do not specify DFLTENTRY as the destination name.

XTPFORMS Form name assigned to the printer definition you want IP PrintWay to use, padded to the right with blanks. Do not use lowercase characters. Also set XTP_ROUTING_KEY_TYPE to '0'.

XTPCOPYS Number of copies of the data set. Specify a number from 1 - 255.

XTP_ADDR_TYPE

Value in field XTPIPADR:

0 = IP address or host name; 1 = URL

XTP_ROUTING_KEY_TYPE

Value in field XTPKEY:

0 = DEST, CLASS, and FORMS; 1 = printer definition name

If you do *not* set any of the flags in field XTPDSFLG, you can change the values in the following fields in the ANFUEXTP control block. IP PrintWay uses the changed values when it transmits the data set. (IP PrintWay ignores changes you make to fields XTPKEY, XTPCLASS, XTPFORMS, and XTPDEST.)

XTPOPTNM Name of components in the Printer Inventory that contain formatting and transmission options. This field is case sensitive and must be padded to the right with blanks.

XTPPQLEN Length of the print queue name in field XTPPRTQU.

XTPPRTQU Name of the target print queue.

XTPIALEN Length of the value in field XTPIPADR.

XTPIPADR Host name or IP address of the target system; or the URL of the target system. Set field XTP_ADDR_TYPE to identify which value is in this field.

XTPRETNS Period of time IP PrintWay is to retain the data set on the JES spool after a successful transmission. See the note on page 95 for the format of this field.

XTPRETNF	Period of time IP PrintWay is to retain the data set on the JES spool after all transmission retries have failed. See the note on page 95 for the format of this field.
XTPRTRYL	Number of times IP PrintWay is to retry transmission of the data set. Specify a number from 0 - 32767.
XTPRTRYT	Period of time IP PrintWay is to wait between retries of the transmission. See the note on page 95 for the format of this field.
XTPCOPYS	Number of copies of the data set. Specify a number from 1 - 255.
XTPIPORT	Port number on the target system.
XTP_ADDR_TYPE	Value in field XTPIPADR: 0 = IP address or host name; 1 = URL
XTP_ROUTING_KEY_TYPE	Value in field XTPKEY: 0 = DEST, CLASS, and FORMS; 1 = printer definition name

Examples

1. To hold the data set on the JES spool for operator intervention, set flag XTPDSHLD=1 in the ANFUEXTP control block.
2. To delete the data set from the JES spool, set flag XTPDSTRM=1 in the ANFUEXTP control block.
3. To request that IP PrintWay obtain routing parameters from a specific printer definition, do the following:
 - Set XTPDSRTG=1 in the ANFUEXTP control block.
 - Specify the printer definition name in field XTPKEY; or specify the class, destination name, and forms name in fields XTPCLASS, XTPDEST, and XTPFORMS.
 - Set XTP_ROUTING_KEY_TYPE, depending on whether you specified the printer definition name or the class, destination, and forms names.
4. To transmit the data set without changing any routing information, simply return control to IP PrintWay.
5. To transmit the data set to:
 - A port number instead of the specified print queue name:
 - Place the port number in XTPIPORT
 - Clear XTPPQLEN
 - A different port number, overriding the previous port number:
 - Place the port number in XTPIPORT
 - A print queue, overriding the port number:
 - Clear XTPIPORT
 - Place the name of the target print queue in field XTPPRTQU
 - Place the length of the target print queue in field XTPPQLEN
 - A different print queue, overriding the previous print queue:
 - Place the name of the target print queue in field XTPPRTQU.
 - Place the length of the target print queue in field XTPPQLEN.

Note: If XTPIPORT, XTPPRTQU and XTPPQLEN are coded in the exit, the XTPIPORT field will be used and the print queue fields will be ignored.

6. To request that IP PrintWay issue a message to the IP PrintWay message-log data set, do the following:
 - Set XTPM2MDS=1 in the ANFUEXTP control block to write the message to the IP PrintWay message-log data set.
 - Place the address of the message in field XTPMSGP. IP PrintWay adds a message ID, ANFM155I, to the beginning of this message.
 - Place the length of the message in field XTPMSGLN.

Refer also to the sample exits distributed with IP PrintWay and described in “Sample Exits” on page 124.

Begin Data Set Exit

The Begin Data Set exit allows you to inspect the transmission options for a data set before the data set is transmitted. In this exit, you can change most of the transmission options, and you can also add a record to the beginning of the data set. You can also inspect the routing parameters for the data set, but you *cannot* change them.

If you request to add a record, you can specify whether or not (1) IP PrintWay should translate the record to ASCII (2) the record contains a carriage control character, and (3) the record contains double-byte characters. You can add more than one record to the beginning of the data set by requesting that IP PrintWay call this exit multiple times for the data set.

The Begin Data Set exit can use the Block Letter program (ANFUBLK), provided with IP PrintWay, to build block letters suitable for printing on a separator page. See “Using the Block Letter Program (ANFUBLK)” on page 122 for more information.

You can also create a message for IP PrintWay to issue either to the operator's console or to the IP PrintWay message-log data set. IP PrintWay assigns the same message ID to each message created by an exit: ANFM155I.

You name the Begin Data Set exit in the printer definition. Therefore, you can write different Begin Data Set exits for different data sets, provided that the data sets use different printer definitions.

You might want to write a Begin Data Set exit to perform one of the following functions:

- Add a printer setup string to the start of the data set. For example, your installation might want to add a setup string for landscape printing.²
- Add a custom separator page. On the separator page, you can print job and data set information contained in the JES job separator page data area (IAZJSPA) or the scheduler work block text unit (SWBTU). Refer to “JES Job Separator Page Data Area (IAZJSPA)” on page 92 and “Scheduler Work Block Text Unit (SWBTU)” on page 93 for more information about these system control blocks.

Input to Begin Data Set Exit

At entry to the Begin Data Set exit, the registers contain the following:

2. If you add a setup string to data, in some cases you may need to redefine the print driver on your workstation. For example, when printing data on an HP Laserjet III Si printer on a Windows NT system, define the HP print driver as a “generic/text only” driver. This prevents the print driver from changing your setup strings to be invalid.

Register 1	Pointer to the address of the ANFUEXTP control block. The ANFUEXTP control block contains addresses of the IAZJSPA and the SWBTU control blocks.
Register 13	Address of an 18-word save area for saving the caller's registers.
Register 14	Return address.
Register 15	Entry-point address.

The following fields in ANFUEXTP are valid when the Begin Data Set exit is called. The format of the ANFUEXTP control block is shown in Figure 26 on page 94.

XTPSAVE	A register save area for use by the exit.
XTPWORK1	Address of a work area previously obtained by an exit. This field contains zeroes if a work area has not been obtained.
XTPWORK2	Address of a second work area previously obtained by an exit. This field contains zeroes if a work area has not been obtained.
XTPDSNAM	Fully qualified name of the data set being processed, padded to the right with blanks.
XTPUSRID	User ID of the job submitter, padded to the right with blanks.
XTPDSFLG	Flags containing data set information:
Flag	Meaning When Set to B'1'.
XTPDSCC	This data set contains carriage control characters.
XTPDSJOB	This is the first data set in the job.
XTPDSJNW	This data set is the JESNEWS data set.
XTPDSEJB	This is the last data set in the job.
XTPDSCON	This data set is part of a concatenation.
XTPRCFLG	Flags containing data set information:
Flag	Meaning When Set to B'1'.
XTPRDFST	This is the first call to the exit for this data set.
XTPPQLEN	Length of the print queue name in field XTPPRTQU.
XTPIALEN	Length of the value in field XTPIPADR.
XTPPRTQU	Name of the target print queue, as specified in the PRTQUEUE JCL parameter, in a printer definition, or by the Routing exit.
XTPIPADR	Host name or IP address of the target system, as specified in the DEST=IP JCL parameter, in a printer definition, or by the Routing exit; or the URL of the target system specified in the printer definition or the Routing exit. Field XTP_ADDR_TYPE identifies which value is in this field.
XTPRETNS	Period of time IP PrintWay is to retain the data set on the JES spool after a successful transmission, as specified in the RETAINS JCL parameter, in a printer definition, or by the Routing exit. IP PrintWay may have set a default value of zeroes. See the note on page 95 for the format of this field.
XTPRETNF	Period of time IP PrintWay is to retain the data set on the JES spool after all transmission retries have failed, as specified in the

RETAINF JCL parameter, in a printer definition, or by the Routing exit. IP PrintWay may have set a default value of zeroes. See the note on page 95 for the format of this field.

XTPRTRYL	Number of times IP PrintWay is to retry transmission of the data set, as specified in the RETRYL JCL parameter, in a printer definition, or by the Routing exit. IP PrintWay may have set a default value of zero.
XTPRTRYT	Period of time IP PrintWay is to wait between retries of the transmission, as specified in the RETRYT JCL parameter, in a printer definition, or by the Routing exit. IP PrintWay may have set a default value of zero. See the note on page 95 for the format of this field.
XTPKEY	If XTP_ROUTING_KEY_TYPE is set to '1', contains the name of the printer definition, as specified in JCL, in the printer definition, or by the Routing exit; the name is padded to the right with blanks. This field is case sensitive.
XTPDEST	Destination name of the data set, as specified in JCL, by the Routing exit, or defaulted by JES, if XTP_ROUTING_KEY_TYPE is set to '0'. The name is padded to the right with blanks. This field contains DFLTNTY if the DEST=IP parameter was specified in the OUTPUT JCL statement without the name of a printer definition.
XTPCLASS	Class of the data set, as specified in JCL, by the Routing exit, or defaulted by JES, if XTP_ROUTING_KEY_TYPE is set to '0'.
XTPFORMS	Form name of the data set, as specified in JCL, by the Routing exit, or defaulted by JES, if XTP_ROUTING_KEY_TYPE is set to '1'. The name is padded to the right with blanks.
XTPRECFM	Record format as indicated in the job file control block (JFCB). Refer to <i>OS/390 MVS Data Areas, Vol 3 (IVT-RCWK)</i> for a description of the JFCRECFM field in the JFCB.
XTPMRECL	Maximum record length
XTPSWBTP	Address of the SWBTU for the data set. See "Scheduler Work Block Text Unit (SWBTU)" on page 93 for more information.
XTPSWBTL	Length of the SWBTU.
XTPJSPAP	Address of the JES job separator page data area for the data set being processed (IAZJSPA). See "JES Job Separator Page Data Area (IAZJSPA)" on page 92 for more information.
XTPCOPYS	Number of copies of the data set, as specified in JCL, by the Routing exit, or defaulted by JES.
XTPIPORT	Port number on the target system, as specified in the PORTNO JCL parameter or in the printer definition.
XTP_ADDR_TYPE	Value in field XTPIPADR: 0 = IP address or host name; 1 = URL
XTP_ROUTING_KEY_TYPE	Value in field XTPKEY: 0 = DEST, CLASS, and FORMS; 1 = printer definition name

XTPOPTNS Transmission options, as specified in the printer definition or the components named in the PRTOPTNS JCL parameter.

Output from Begin Data Set Exit

At return, restore the contents of all registers. Do not set a return code in register 15.

The exit can set the following fields in the ANFUEXTP control block:

XTPWORK1	Address of a work area obtained by this exit. This address is passed to every exit except the Message exit.												
XTPWORK2	Address of a second work area obtained by this exit. This address is passed to every exit except the Message exit.												
XTPRCFLG	Flags to request that IP PrintWay add either the original record or a new record and to specify characteristics of the new record. IP PrintWay sets these flags to B'0' before calling the exit. To transmit the data set without adding a record to the beginning of the data set, do <i>not</i> set any of these flags to B'1'.												
<table> <tr> <th>Flag</th><th>Meaning When Set to B'1'.</th></tr> <tr> <td>XTPRCTRN</td><td>Translate the new record to ASCII. IP PrintWay uses the single-byte or double-byte translation option specified in the transmission options.</td></tr> <tr> <td>XTPRCCC</td><td>The new record contains a carriage control character in column 1.</td></tr> <tr> <td>XTPRLAST</td><td>Do not call this exit again to add another record to the beginning of this data set.</td></tr> <tr> <td>XTPRCEXT</td><td>Add the record whose address is in field XTPERPTR.</td></tr> <tr> <td>XTPRCDBL</td><td>The new record contains double-byte characters.</td></tr> </table>		Flag	Meaning When Set to B'1'.	XTPRCTRN	Translate the new record to ASCII. IP PrintWay uses the single-byte or double-byte translation option specified in the transmission options.	XTPRCCC	The new record contains a carriage control character in column 1.	XTPRLAST	Do not call this exit again to add another record to the beginning of this data set.	XTPRCEXT	Add the record whose address is in field XTPERPTR.	XTPRCDBL	The new record contains double-byte characters.
Flag	Meaning When Set to B'1'.												
XTPRCTRN	Translate the new record to ASCII. IP PrintWay uses the single-byte or double-byte translation option specified in the transmission options.												
XTPRCCC	The new record contains a carriage control character in column 1.												
XTPRLAST	Do not call this exit again to add another record to the beginning of this data set.												
XTPRCEXT	Add the record whose address is in field XTPERPTR.												
XTPRCDBL	The new record contains double-byte characters.												
XTPMSGFL	Flags indicating the destination of a message created by this exit. IP PrintWay sets these flags to B'0' before calling the exit. Set both flags to B'1' to issue the message to both destinations. Place the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.												
<table> <tr> <th>Flag</th><th>Meaning When Set to B'1'.</th></tr> <tr> <td>XTPM2CON</td><td>Issue the message to the operator's console.</td></tr> <tr> <td>XTPM2MDS</td><td>Issue the message to the IP PrintWay message-log data set.</td></tr> </table>		Flag	Meaning When Set to B'1'.	XTPM2CON	Issue the message to the operator's console.	XTPM2MDS	Issue the message to the IP PrintWay message-log data set.						
Flag	Meaning When Set to B'1'.												
XTPM2CON	Issue the message to the operator's console.												
XTPM2MDS	Issue the message to the IP PrintWay message-log data set.												
XTPMSGLN	Length of the message whose address is in field XTPMSGP.												
XTPMSGP	Address of the message created by this exit. IP PrintWay adds a message ID, ANFM155I, to the beginning of this message.												
XTPERPTR	Address of the record to be added.												
XTPERLEN	Length of the record to be added.												
XTPOPTNS	The transmission options. Refer to the ANFUEXTP macro for the field names. If you change a value in one of the fields, also set the length of the value in the length field that precedes that field.												

Examples

1. To add one record to the beginning of the data set, set the following fields in ANFUEXTP:
 - Set flag XTPRCEXT=1 to request that IP PrintWay add a record.
 - Place the address of the new record in field XTPERPTR.
 - Place the length of the new record in field XTPERLEN.
 - Set flags to indicate the characteristics of the record you are adding:
 - Set flag XTPRCTRN=1 if you want IP PrintWay to translate the record to ASCII.
 - Set flag XTPRCCC=1 if the record contains a carriage-control character.
 - Set flag XTPRCDBL=1 if the record contains double-byte characters.
 - Set flag XTPRLAST=1 to indicate this is the last record you are adding.
2. To specify a transmission option, set the following fields in the ANFUEXTP control block:
 - Specify the value in the field for that transmission option in XTPOPTNS.
 - Specify the length of the value in the length field for that transmission option.
3. To transmit the data set with no changes, simply return control to IP PrintWay.
4. To request that IP PrintWay issue a message to the IP PrintWay message-log data set, do the following:
 - Set XTPM2MDS=1 in the ANFUEXTP control block to write the message to the IP PrintWay message-log data set.
 - Place the address of the message in field XTPMSGP. IP PrintWay adds a message ID, ANFM155I, to the beginning of this message.
 - Place the length of the message in field XTPMSGLN.

Refer also to the sample exits distributed with IP PrintWay and described in “Sample Exits” on page 124.

Record Exit

The Record exit allows you to inspect each record in a data set before the data set is transmitted. You can add one or more records, replace a record, or delete a record. You can also inspect the routing parameters and the transmission options for the data set, but you *cannot* change them.

If you request to add or replace a record, you can specify whether or not (1) IP PrintWay should translate the new record to ASCII, (2) the new record contains a carriage control character, and (3) the new record contains double-byte characters. You can add more than one record at the same location in the data set by requesting that IP PrintWay call the exit multiple times.

You can also create a message for IP PrintWay to issue either to the operator's console or to the IP PrintWay message-log data set. IP PrintWay assigns the same message ID to each message created by an exit: ANFM155I.

You name the Record exit in the printer definition. Therefore, you can write different Record exits for different data sets, provided that the data sets use different printer definitions.

Note: IP PrintWay does not call the Record exit for records added by the Begin Data Set exit or the End Data Set exit.

Input to Record Exit

At entry to the Record exit, the registers contain the following:

Register 1	Pointer to the address of the ANFUEXTP control block. The ANFUEXTP control block contains addresses of the IAZJSPA and the SWBTU control blocks.
Register 13	Address of an 18-word save area for saving the caller's registers.
Register 14	Return address.
Register 15	Entry-point address.

The following fields in ANFUEXTP are valid when the Record exit is called. The format of the ANFUEXTP control block is shown in Figure 26 on page 94.

XTPSAVE	A register save area for use by the exit.						
XTPWORK1	Address of a work area previously obtained by an exit. This field contains zeroes if a work area has not been obtained.						
XTPWORK2	Address of a second work area previously obtained by an exit. This field contains zeroes if a work area has not been obtained.						
XTPDSNAM	Fully-qualified name of the data set being processed, padded to the right with blanks.						
XTPUSRID	User ID of the job submitter, padded to the right with blanks.						
XTPDSFLG	Flags containing data set information: <table><tr><th>Flag</th><th>Meaning When Set to B'1'.</th></tr><tr><td>XTPDSCC</td><td>This data set contains carriage control characters.</td></tr><tr><td>XTPDSJNW</td><td>This data set is the JESNEWS data set.</td></tr></table>	Flag	Meaning When Set to B'1'.	XTPDSCC	This data set contains carriage control characters.	XTPDSJNW	This data set is the JESNEWS data set.
Flag	Meaning When Set to B'1'.						
XTPDSCC	This data set contains carriage control characters.						
XTPDSJNW	This data set is the JESNEWS data set.						
XTPRCFLG	Flags containing record information: <table><tr><th>Flag</th><th>Meaning When Set to B'1'.</th></tr><tr><td>XTPRDFST</td><td>This is the first call to the exit for this record.</td></tr></table>	Flag	Meaning When Set to B'1'.	XTPRDFST	This is the first call to the exit for this record.		
Flag	Meaning When Set to B'1'.						
XTPRDFST	This is the first call to the exit for this record.						
XTPPRPTR	Address of the record currently in the data set (original record).						
XTPPRLEN	Length of record whose address is in field XTPPRPTR.						
XTPPQLEN	Length of the print queue name in field XTPPRTQU.						
XTPIALEN	Length of the value in field XTPIPADR.						
XTPPRTQU	Name of the target print queue, as specified in the PRTQUEUE JCL parameter, in a printer definition, or by the Routing exit.						
XTPIPADR	Host name or IP address of the target system, as specified in the DEST=IP JCL parameter, in a printer definition, or by the Routing exit; or the URL of the target system specified in the printer definition or the Routing exit. Field XTP_ADDR_TYPE identifies which value is in this field.						
XTPRETNS	Period of time IP PrintWay is to retain the data set on the JES spool after a successful transmission, as specified in the RETAINS JCL parameter, in a printer definition, or by the Routing exit. IP PrintWay may have set a default value of zeroes. See the note on page 95 for the format of this field.						
XTPRETNF	Period of time IP PrintWay is to retain the data set on the JES spool after all transmission retries have failed, as specified in the						

RETAINF JCL parameter, in a printer definition, or by the Routing exit. IP PrintWay may have set a default value of zeroes. See the note on page 95 for the format of this field.

XTPRTRYL	Number of times IP PrintWay is to retry transmission of the data set, as specified in the RETRYL JCL parameter, in a printer definition, or by the Routing exit. IP PrintWay may have set a default value of zero.
XTPRTRYT	Period of time IP PrintWay is to wait between retries of the transmission, as specified in the RETRYT JCL parameter, in a printer definition, or by the Routing exit. IP PrintWay may have set a default value of zero. See the note on page 95 for the format of this field.
XTPKEY	If XTP_ROUTING_KEY_TYPE is set to '1', contains the name of the printer definition, as specified in JCL, the printer definition, or by the Routing exit; the name is padded to the right with blanks.
XTPDEST	If XTP_ROUTING_KEY_TYPE is set to '0', contains the destination name of the data set, as specified in JCL, by the Routing exit, or defaulted by JES; the name is padded to the right with blanks. This field contains DFLTENTRY if the DEST=IP parameter was specified in the OUTPUT JCL statement without the name of a printer definition. This field contains blanks if the DEST field in the printer definition is not specified.
XTPCLASS	If XTP_ROUTING_KEY_TYPE is set to '0', contains the class of the data set, as specified in JCL, by the Routing exit, or defaulted by JES. This field contains blanks if the CLASS field in the printer definition is not specified.
XTPFORMS	If XTP_ROUTING_KEY_TYPE is set to '0', contains the form name of the data set, as specified in JCL, by the Routing exit, or defaulted by JES; the name is padded to the right with blanks. This field contains blanks if the FORMS field in the printer definition is not specified.
XTPSWBTP	Address of the SWBTU for the data set. See “Scheduler Work Block Text Unit (SWBTU)” on page 93 for more information.
XTPSWBTL	Length of the SWBTU.
XTPJSPAP	Address of the JES job separator page data area for the data set being processed (IAZJSPA). See “JES Job Separator Page Data Area (IAZJSPA)” on page 92 for more information.
XTIPIPORT	Port number on the target system, as specified in the PORTNO JCL parameter or in the printer definition.
XTP_ADDR_TYPE	Value in field XTIPIADR: 0 = IP address or host name; 1 = URL
XTP_ROUTING_KEY_TYPE	Value in field XTPKEY: 0 = DEST, CLASS, and FORMS; 1 = printer definition name
XTPOPTNS	The transmission options. Refer to the ANFUEXTTP macro for the field names. If you change a value in one of the fields, also set the length of the value in the length field that precedes that field.

Output from Record Exit

At return, restore the contents of all registers. Do not set a return code in register 15.

The exit can set the following fields in the ANFUEXTP control block:

XTPWORK1	Address of a work area obtained by this exit. This address is passed to every exit except the Message exit.														
XTPWORK2	Address of a second work area obtained by this exit. This address is passed to every exit except the Message exit.														
XTPRCFLG	Flags to request that IP PrintWay add a record and to specify characteristics of a new record. IP PrintWay sets these flags to B'0' before calling the exit. <table><tr><th>Flag</th><th>Meaning When Set to B'1'.</th></tr><tr><td>XTPRCTRN</td><td>Translate the new record to ASCII. IP PrintWay uses the single-byte or double-byte translation option specified in the transmission options. This flag does <i>not</i> apply to the original record.</td></tr><tr><td>XTPRCCC</td><td>The new record contains a carriage control character in column 1. This flag does <i>not</i> apply to the original record.</td></tr><tr><td>XTPRCORG</td><td>Add the original record to the data set.</td></tr><tr><td>XTPRLAST</td><td>Do <i>not</i> call this exit again for this record. If neither field XTPRCORG nor XTPRCEXT are set, IP PrintWay ignores this flag and does <i>not</i> call the exit again for this record.</td></tr><tr><td>XTPRCEXT</td><td>Add the new record whose address is in field XTPERPTR. Do <i>not</i> set this flag and flag XTPRCORG at the same time. If flag XTPRCORG is set, IP PrintWay ignores this flag.</td></tr><tr><td>XTPRCDBL</td><td>The new record contains double-byte characters. This flag does <i>not</i> apply to the original record.</td></tr></table>	Flag	Meaning When Set to B'1'.	XTPRCTRN	Translate the new record to ASCII. IP PrintWay uses the single-byte or double-byte translation option specified in the transmission options. This flag does <i>not</i> apply to the original record.	XTPRCCC	The new record contains a carriage control character in column 1. This flag does <i>not</i> apply to the original record.	XTPRCORG	Add the original record to the data set.	XTPRLAST	Do <i>not</i> call this exit again for this record. If neither field XTPRCORG nor XTPRCEXT are set, IP PrintWay ignores this flag and does <i>not</i> call the exit again for this record.	XTPRCEXT	Add the new record whose address is in field XTPERPTR. Do <i>not</i> set this flag and flag XTPRCORG at the same time. If flag XTPRCORG is set, IP PrintWay ignores this flag.	XTPRCDBL	The new record contains double-byte characters. This flag does <i>not</i> apply to the original record.
Flag	Meaning When Set to B'1'.														
XTPRCTRN	Translate the new record to ASCII. IP PrintWay uses the single-byte or double-byte translation option specified in the transmission options. This flag does <i>not</i> apply to the original record.														
XTPRCCC	The new record contains a carriage control character in column 1. This flag does <i>not</i> apply to the original record.														
XTPRCORG	Add the original record to the data set.														
XTPRLAST	Do <i>not</i> call this exit again for this record. If neither field XTPRCORG nor XTPRCEXT are set, IP PrintWay ignores this flag and does <i>not</i> call the exit again for this record.														
XTPRCEXT	Add the new record whose address is in field XTPERPTR. Do <i>not</i> set this flag and flag XTPRCORG at the same time. If flag XTPRCORG is set, IP PrintWay ignores this flag.														
XTPRCDBL	The new record contains double-byte characters. This flag does <i>not</i> apply to the original record.														
XTPMSGFL	Flags indicating the destination of a message created by this exit. Set both flags to B'1' to issue the message to both destinations. Place the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN. <table><tr><th>Flag</th><th>Meaning When Set to B'1'.</th></tr><tr><td>XTPM2CON</td><td>Issue the message to the operator's console.</td></tr><tr><td>XTPM2MDS</td><td>Issue the message to the IP PrintWay message-log data set.</td></tr></table>	Flag	Meaning When Set to B'1'.	XTPM2CON	Issue the message to the operator's console.	XTPM2MDS	Issue the message to the IP PrintWay message-log data set.								
Flag	Meaning When Set to B'1'.														
XTPM2CON	Issue the message to the operator's console.														
XTPM2MDS	Issue the message to the IP PrintWay message-log data set.														
XTPERPTR	Address of the record to be added.														
XTPERLEN	Length of the record to be added.														
XTPPRPTR	Address of the original record in the data set. You can change the contents of the original record.														
XTPPRLEN	Length of the original record whose address is in field XTPPRPTR.														
XTPMSGLN	Length of the message whose address is in field XTPMSGP.														

XTPMSGP Address of the message created by this exit. IP PrintWay adds a message ID, ANFM155I, to the beginning of this message.

Examples

1. To add a new record after the original record in the data set, set the following fields in ANFUExTP:
 - Set flag XTPRCORG=1 to request that IP PrintWay transmit the original record.
 - Set flag XTPRLAST=0 to indicate that IP PrintWay is to call the exit again to add another record.

On the next call to the Record Exit, set the following fields in ANFUExTP:

- Set flag XTPRCORG=0 to request that IP PrintWay not add the original record.
 - Set flag XTPRCEXT=1 to request that IP PrintWay add a new record.
 - Place the address of the new record in field XTPERPTR.
 - Place the length of the new record in field XTPERLEN.
 - Set flags to indicate the characteristics of the new record:
 - Set flag XTPRCTRN=1 if you want IP PrintWay to translate the record to ASCII.
 - Set flag XTPRCCC=1 if the record contains a carriage-control character.
 - Set flag XTPRCDBL=1 if the record contains double-byte characters.
 - Set flag XTPRLAST=1 to indicate this is the last record you are adding.
2. To replace the original record, set the following fields in ANFUExTP:
 - Set flag XTPRCORG=1 to request that IP PrintWay add the original record.
 - Modify the record whose address is in field XTPPRPTR; however, do not increase the length of the record in the original location. To make the record longer, create the record in another location and place the address of the record in field XTPPRPTR.
 - If the length of the record has changed, place the new record length in XTPPRLN.
 - Set flag XTPRLAST=1 to indicate this is the last record to be added.
 3. To make *no* changes to the original record:
 - Set flag XTPRCORG=1 to request that IP PrintWay add the original record.
 - Set flag XTPRLAST=1 to indicate this is the last record to be added.
 4. To delete the original record, simply return control to IP PrintWay.
 5. To request that IP PrintWay issue a message to the IP PrintWay message-log data set, do the following:
 - Set XTPM2MDS=1 in the ANFUExTP control block to write the message to the IP PrintWay message-log data set.
 - Place the address of the message in field XTPMSGP. IP PrintWay adds a message ID, ANFM155I, to the beginning of this message.
 - Place the length of the message in field XTPMSGLN.

Refer also to the sample exits distributed with IP PrintWay and described in “Sample Exits” on page 124.

End Data Set Exit

The End Data Set exit allows you to add one or more records to the end of a data set. You can also inspect the routing parameters and the transmission options for the data set, but you *cannot* change them.

If you request to add a record, you can specify whether or not (1) IP PrintWay should translate the new record to ASCII, (2) the new record contains a carriage control character, and (3) the new record contains double-byte characters. You can add more than one record by requesting that IP PrintWay call the exit multiple times.

You can also create a message for IP PrintWay to issue either to the operator's console or to the IP PrintWay message-log data set. IP PrintWay assigns the same message ID to each message created by an exit: ANFM155I.

The End Data Set exit can use the Block Letter program (ANFUBLK), provided with IP PrintWay, to build block letters suitable for printing on a separator page. See "Using the Block Letter Program (ANFUBLK)" on page 122 for more information.

You name the End Data Set exit in the printer definition. Therefore, you can write different End Data Set exits for different data sets, provided that the data sets use different options entries.

You might want to write an End Data Set exit to perform one of the following functions:

- Append fonts to be transmitted to the printer.
- Add a custom separator page. On the separator page you can print job and data set information contained in the JES job separator page data area (IAZJSPA) or the scheduler work block text unit (SWBTU). See "JES Job Separator Page Data Area (IAZJSPA)" on page 92 and "Scheduler Work Block Text Unit (SWBTU)" on page 93 for more information about these system control blocks.

Input to End Data Set Exit

At entry to the End Data Set exit, the registers contain the following:

Register 1	Pointer to the address of the ANFUEXTP control block. The ANFUEXTP control block contains addresses of the IAZJSPA and the SWBTU control blocks.
Register 13	Address of an 18-word save area for saving the caller's registers.
Register 14	Return address.
Register 15	Entry-point address.

The following fields in ANFUEXTP are valid when the End Data Set exit is called. The format of the ANFUEXTP control block is shown in Figure 26 on page 94.

XTPSAVE	A register save area for use by the exit.
XTPWORK1	Address of a work area previously obtained by an exit. This field contains zeroes if a work area has not been obtained.
XTPWORK2	Address of a second work area previously obtained by an exit. This field contains zeroes if a work area has not been obtained.
XTPDSNAM	Fully qualified name of the data set being processed, padded to the right with blanks.

XTPUSRID	User ID of the job submitter, padded to the right with blanks.
XTPDSFLG	Flags containing data set information:
Flag	Meaning When Set to B'1'.
XTPDSCC	This data set contains carriage control characters.
XTPDSJOB	This is the first data set in the job.
XTPDSJNW	This data set is the JESNEWS data set.
XTPDSEJB	This is the last data set in the job.
XTPDSCON	This data set is part of a concatenation.
XTPRCFLG	Flags containing data set information:
Flag	Meaning When Set to B'1'.
XTPRDFST	This is the first call to the exit for this data set.
XTPPQLEN	Length of the print queue name in field XTPPRTQU.
XTPIALEN	Length of the value in field XTPIPADR.
XTPPRTQU	Name of the target print queue, as specified in the PRTQUEUE JCL parameter, in a printer definition, or by the Routing exit.
XTPIPADR	Host name or IP address of the target system, as specified in the DEST=IP JCL parameter, in a printer definition, or by the Routing exit; or the URL of the target system specified in the printer definition or the Routing exit. Field XTP_ADDR_TYPE identifies which value is in this field.
XTPRETNS	Period of time IP PrintWay is to retain the data set on the JES spool after a successful transmission, as specified in the RETAINS JCL parameter, in a printer definition, or by the Routing exit. IP PrintWay may have set a default value of zeroes. See the note on page 95 for the format of this field.
XTPRETNF	Period of time IP PrintWay is to retain the data set on the JES spool after all transmission retries have failed, as specified in the RETAINF JCL parameter, in a printer definition, or by the Routing exit. IP PrintWay may have set a default value of zeroes. See the note on page 95 for the format of this field.
XTPRTRYL	Number of times IP PrintWay is to retry transmission of the data set, as specified in the RETRYL JCL parameter, in a printer definition, or by the Routing exit. IP PrintWay may have set a default value of zero.
XTPRTRYT	Period of time IP PrintWay is to wait between retries of the transmission, as specified in the RETRYT JCL parameter, in a printer definition, or by the Routing exit. IP PrintWay may have set a default value of zero. See the note on page 95 for the format of this field.
XTPKEY	If XTP_ROUTING_KEY_TYPE is set to '1', contains the name of the printer definition, as specified in JCL, the printer definition, or by the Routing exit; the name is padded to the right with blanks.
XTPDEST	If XTP_ROUTING_KEY_TYPE is set to '0', contains the destination name of the data set, as specified in JCL, by the Routing exit, or defaulted by JES; the name is padded to the right with blanks. This field contains DFLTENTRY if the DEST=IP parameter was specified in

the OUTPUT JCL statement without the name of a printer definition. This field contains blanks if the DEST field in the printer definition is not specified.

XTPCLASS	If XTP_ROUTING_KEY_TYPE is set to '0', contains the class of the data set, as specified in JCL, by the Routing exit, or defaulted by JES. This field contains blanks if the CLASS field in the printer definition is not specified.
XTPFORMS	If XTP_ROUTING_KEY_TYPE is set to '0', contains the form name of the data set, as specified in JCL, by the Routing exit, or defaulted by JES; the name is padded to the right with blanks. This field contains blanks if the FORMS field in the printer definition is not specified.
XTPRECFM	Record format as indicated in the job file control block (JFCB). Refer to <i>OS/390 MVS Data Areas, Vol 3 (IVT-RCWK)</i> for a description of the JFCRECFM field in the JFCB.
XTPMRECL	Maximum record length
XTPSWBTP	Address of the SWBTU for the data set. See “Scheduler Work Block Text Unit (SWBTU)” on page 93 for more information.
XTPSWBTL	Length of the SWBTU.
XTPJSPAP	Address of the JES job separator page data area for the data set being processed (IAZJSPA). See “JES Job Separator Page Data Area (IAZJSPA)” on page 92 for more information.
XTPCOPYS	Number of copies of the data set, as specified in JCL, by the Routing exit, or defaulted by JES.
XTPIPORT	Port number on the target system, as specified in the PORTNO JCL parameter or in the printer definition.
XTPOPTNS	The transmission options. Refer to the ANFUEXTP macro for the field names. If you change a value in one of the fields, also set the length of the value in the length field that precedes that field.

Output from End Data Set Exit

At return, restore the contents of all registers. Do not set a return code in register 15.

The exit can set the following fields in the ANFUEXTP control block:

XTPWORK1	Address of a work area obtained by this exit. This address is passed to every exit except the Message exit.
XTPWORK2	Address of a second work area obtained by this exit. This address is passed to every exit except the Message exit.
XTPRCFLG	Flags to request that IP PrintWay add a record and to specify characteristics of the new record. IP PrintWay sets these flags to B'0' before calling the exit. To transmit the data set without adding a record to the end of the data set, do <i>not</i> set any of these flags.
	Flag Meaning When Set to B'1'.
XTPRCTRN	Translate the new record to ASCII. IP PrintWay uses the single-byte or double-byte translation option specified in the transmission options.

XTPRCCC	The new record contains a carriage control character in column 1.
XTPRLAST	Do not call this exit again to add another record to the end of this data set.
XTPRCEXT	Add the record whose address is in field XTPERPTR.
XTPRCDBL	The new record contains double-byte characters.
XTPMSGFL	Flags indicating the destination of a message created by this exit. IP PrintWay sets these flags to B'0' before calling the exit. Set both flags to B'1' to issue the message to both destinations. Place the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.
Flag	Meaning When Set to B'1'.
XTPM2CON	Issue the message to the operator's console.
XTPM2MDS	Issue the message to the IP PrintWay message-log data set.
XTPERPTR	Address of the record to be added.
XTPERLEN	Length of the record to be added.
XTPMSGLN	Length of the message whose address is in field XTPMSGP.
XTPMSGP	Address of the message created by this exit. IP PrintWay adds a message ID, ANFM155I, to the beginning of this message.

Examples

- To add one record to the end of the data set, set the following fields in ANFUEXTTP:
 - Set flag XTPRCEXT=1 to request that IP PrintWay add a record.
 - Place the address of the new record in field XTPERPTR.
 - Place the length of the new record in field XTPERLEN.
 - Set flags to indicate the characteristics of the record you are adding:
 - Set flag XTPRCTR=1 if you want IP PrintWay to translate the record to ASCII.
 - Set flag XTPRCCC=1 if the record contains a carriage-control character.
 - Set flag XTPRCDBL=1 if the record contains double-byte characters.
 - Set flag XTPRLAST=1 to indicate this is the last record you are adding.
- To transmit the data set with no changes, simply return control to IP PrintWay.
- To request that IP PrintWay issue a message to the IP PrintWay message-log data set, do the following:
 - Set XTPM2MDS=1 in the ANFUEXTTP control block to write the message to the IP PrintWay message-log data set.
 - Place the address of the message in field XTPMSGP. IP PrintWay adds a message ID, ANFM155I, to the beginning of this message.
 - Place the length of the message in field XTPMSGLN.

Refer also to the sample exits distributed with IP PrintWay and described in "Sample Exits" on page 124.

SMF Exit (ANFUXSMF)

The SMF exit allows you to replace or suppress the system management facilities (SMF) type-6 records written by IP PrintWay. SMF is an optional control program that collects information to assist you in evaluating system use. If you do not write an SMF exit, IP PrintWay writes a standard SMF type-6 record for each data set transmitted to the target print queue.

For more information about the SMF type-6 record that IP PrintWay writes, refer to *OS/390 Infoprint Server Operation and Administration*. For more information about SMF and about macro IFASMFR, which maps the SMF type 6-record, refer to *OS/390 MVS System Management Facilities (SMF)*.

You can also request that IP PrintWay issue a message produced in the exit to the operator's console or to the IP PrintWay message-log data set. IP PrintWay assigns the same message ID to all messages produced by any exit: ANFM155I.

IP PrintWay calls the same SMF exit for all data sets. You can install different SMF exits for different IP PrintWay FSSs. See "Installing Exits" on page 124 for information.

You might want to write an SMF exit to suppress the SMF type-6 record for JESNEWS data sets or to include additional information in the SMF type-6 record.

Note: Fields in macro IFASMFR that are provided by SVC 83 are not available to this exit.

Input to SMF Exit

At entry to the SMF exit, the registers contain the following:

Register 1	Pointer to the address of the ANFUEXTP control block. The ANFUEXTP control block contains addresses of the IAZJSPA and the SWBTU control blocks.
Register 13	Address of an 18-word save area for saving the caller's registers.
Register 14	Return address.
Register 15	Entry-point address.

The following fields in ANFUEXTP are valid when the SMF exit is called. The format of the ANFUEXTP control block is shown in Figure 26 on page 94.

XTPSAVE	A register save area for use by the exit.
XTPWORK1	Address of a work area previously obtained by an exit. This field contains zeroes if a work area has not been obtained.
XTPWORK2	Address of a second work area previously obtained by an exit. This field contains zeroes if a work area has not been obtained.
XTPDSNAM	Fully qualified name of the data set being processed, padded to the right with blanks.
XTPUSRID	User ID of the job submitter, padded to the right with blanks.
XTPDSFLG	Flags containing data set information:
Flag	Meaning When Set to B'1'.
XTPDSJOB	This is the first data set in the job.
XTPDSJNW	This data set is the JESNEWS data set.

XTPDSEJB	This is the last data set in the job.
XTPDSCON	This data set is part of a concatenation.
XTPPRPTR	Address of the SMF record for the data set.
XTPPRLEN	Length of the SMF record for the data set.
XTPKEY	If XTP_ROUTING_KEY_TYPE is set to '1', contains the name of the printer definition, as specified in JCL, the printer definition, or by the Routing exit; the name is padded to the right with blanks.
XTPDEST	If XTP_ROUTING_KEY_TYPE is set to '0', contains the destination name of the data set, as specified in JCL, by the Routing exit, or defaulted by JES; the name is padded to the right with blanks. This field contains DFLTENTRY if the DEST=IP parameter was specified in the OUTPUT JCL statement without the name of a printer definition. This field contains blanks if the DEST field in the printer definition is not specified.
XTPCLASS	If XTP_ROUTING_KEY_TYPE is set to '0', contains the class of the data set, as specified in JCL, by the Routing exit, or defaulted by JES. This field contains blanks if the CLASS field in the printer definition is not specified.
XTPFORMS	If XTP_ROUTING_KEY_TYPE is set to '0', contains the form name of the data set, as specified in JCL, by the Routing exit, or defaulted by JES; the name is padded to the right with blanks. This field contains blanks if the FORMS field in the printer definition is not specified.
XTPSWBTP	Address of the SWBTU for the data set. See “Scheduler Work Block Text Unit (SWBTU)” on page 93 for more information.
XTPSWBTL	Length of the SWBTU.
XTPJSPAP	Address of the JES job separator page data area for the data set being processed (IAZJSPA). See “JES Job Separator Page Data Area (IAZJSPA)” on page 92 for more information.
XTPCOPYS	Number of copies of the data set, as specified in JCL, by the Routing exit, or defaulted by JES.
XTP_ROUTING_KEY_TYPE	Value in field XTPKEY: 0 = DEST, CLASS, and FORMS; 1 = printer definition name

Output from SMF Exit

At return, restore the contents of all registers. Do not set a return code in register 15.

The exit can set the following fields in the ANFUEXTP control block:

XTPWORK1	Address of a work area obtained by this exit. This address is passed to every exit except the Message exit.
XTPWORK2	Address of a second work area obtained by this exit. This address is passed to every exit except the Message exit.
XTPRCFLG	Flag to request that IP PrintWay write or suppress the SMF record. IP PrintWay sets this flag to B'0' before calling the exit.
Flag	Meaning When Set to B'1'.

XTPRCORG	Write the SMF record; otherwise, IP PrintWay suppresses the record.
XTPMSGFL	Flags indicating the destination of a message created by this exit. IP PrintWay sets these flags to B'0' before calling the exit. Set both flags to B'1' to issue the message to both destinations. Place the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.
Flag	Meaning When Set to B'1'.
XTPM2CON	Issue the message to the operator's console.
XTPM2MDS	Issue the message to the IP PrintWay message-log data set.
XTPPRPTR	Address of the SMF type-6 record to be written for the data set. You can change the contents and length of the original record.
XTPPRLN	Length of the SMF type-6 record whose address is in field XTPPRPTR.
XTPMSGLN	Length of the message whose address is in field XTPMSGP.
XTPMSGP	Address of the message created by this exit. IP PrintWay adds a message ID, ANFM155I, to the beginning of this message.

Examples

- To replace the SMF record, set the following fields in ANFUEXIT:
 - Set flag XTPRCORG=1 to request that IP PrintWay transmit the original record.
 - Modify the SMF record whose address is in field XTPPRPTR.
 - If the length of the SMF record has changed, place the new length in field XTPPRLN.
- To write the original SMF record, with no changes, set the following field in ANFUEXIT:
 - Set flag XTPRCORG=1 to request that IP PrintWay transmit the original SMF record.
- To suppress the SMF record, simply return control to IP PrintWay.
- To request that IP PrintWay issue a message to the IP PrintWay message-log data set, do the following:
 - Set XTPM2MDS=1 in the ANFUEXIT control block to write the message to the IP PrintWay message-log data set.
 - Place the address of the message in field XTPMSGP. IP PrintWay adds a message ID, ANFM155I, to the beginning of this message.
 - Place the length of the message in field XTPMSGLN.

Refer also to the sample exits distributed with IP PrintWay and described in "Sample Exits" on page 124.

Message Exit (ANFUXMSG)

The Message exit allows you to either (1) suppress or (2) modify the ID or text of any message that IP PrintWay issues to the IP PrintWay message-log data set, including messages created by another IP PrintWay exit.

IP PrintWay calls the same Message exit for all data sets. You can install different Message exits for different IP PrintWay FSSs. See “Installing Exits” on page 124 for information.

Input to Message Exit

At entry to the Message exit, the registers contain the following:

Register 1	Pointer to the address of the ANFUExTP control block.
Register 13	Address of an 18-word save area for saving the caller's registers.
Register 14	Return address.
Register 15	Entry-point address.

The following fields in ANFUExTP are valid when the Message exit is called. The format of the ANFUExTP control block is shown in Figure 26 on page 94.

XTPSAVE	A register save area for use by the exit.
XTPMSGLN	Length of the message whose address is in field XTPMSGP.
XTPMSGP	Address of the message to be issued. This message has a message ID, followed by one blank and the message text.

Output from Message Exit

At return, restore the contents of all registers. Do not set a return code in register 15.

The exit can set the following fields in the ANFUExTP control block:

XTPMSGFL	Flag indicating whether or not IP PrintWay is to issue the message. IP PrintWay sets this flag to B'0' before calling the exit.				
<table> <tr> <th>Flag</th><th>Meaning When Set to B'1'.</th></tr> <tr> <td>XTPM2MDS</td><td>Issue the message to the IP PrintWay message-log data set.</td></tr> </table>		Flag	Meaning When Set to B'1'.	XTPM2MDS	Issue the message to the IP PrintWay message-log data set.
Flag	Meaning When Set to B'1'.				
XTPM2MDS	Issue the message to the IP PrintWay message-log data set.				
XTPMSGLN	Length of the message whose address is in field XTPMSGP.				
XTPMSGP	Address of the message to be issued. This message should have a message ID, followed by the message text.				

Examples

- To write the message unchanged to the IP PrintWay message-log data set, set the following fields in ANFUExTP:
 - Set flag XTPM2MDS=1 to request that IP PrintWay write the message to the IP PrintWay message-log data set.
- To suppress the message, simply return control to IP PrintWay.
- To replace the message, set the following fields in ANFUExTP:
 - Set XTPM2MDS=1 in the ANFUExTP control block to write the message to the IP PrintWay message-log data set.
 - Modify the message whose address is in field XTPMSGP. Alternatively, place the address of another message in field XTPMSGP.
 - If the length of the message has changed, place the new length in field XTPMSGLN.

Refer also to the sample exits distributed with IP PrintWay and described in “Sample Exits” on page 124.

Response Notification Exit (ANFUXRSP)

The Response Notification Exit allows you to take action, based on the success or failure of the transmission, as indicated by the response notification code. Such actions might include the generation of user-defined messages for either the operator's console or the IP PrintWay message-log data set, or both. Actions outside of IP PrintWay also can be initiated, based on the success of the transmission, by calling modules that are external to IP PrintWay.

IP PrintWay calls the same response exit for all data set transmissions. You can install different response exits for different IP PrintWay FSSs. See "Installing Exits" on page 124 for information.

Input to the Response Notification Exit

Register 1	Pointer to the address of the ANFUEXTP control block.
Register 13	Address of an 18-word save area for saving the caller's registers.
Register 14	Return address.
Register 15	Entry point address.

The following input fields in ANFUEXTP are valid when the Response Notification exit is called. The format of the ANFUEXTP control block is shown in Figure 26 on page 94.

XTPSAVE	A register save area for use by this exit.
XTPWORK1	Address of a work area previously obtained by an exit. This field contains zeros if this work area has not been obtained.
XTPWORK2	Address of a second work area previously obtained by an exit. This field contains zeros if this work area has not been obtained.
XTPDSNAM	Fully qualified name of the data set being processed, padded to the right with blanks.
XTPUSRID	User ID of the job submitter, padded to the right with blanks.
XTPSRCFL	Flags indicating whether a parameter was specified on the output JCL statement for the data set. If the flag is set to 1, the indicated parameter was specified on the output JCL. If the flag is set to 0, the JCL parameter was not specified. Instead, IP PrintWay obtained the value from a printer definition. If the DEST=IP parameter was specified in JCL, IP PrintWay obtained the value from the PrintWay default printer definition. If no PrintWay default printer definition exists, IP PrintWay set a default value.
Flag	Meaning when set to 1.
XTPOPTS	The PRTOPTNS parameter specified the name of components in the Printer Inventory.
XTPSIPAD	The DEST=IP parameter specified the host name or IP address of the target system.
XTPSPRTQ	The PRTQUEUE parameter specified the name of the target print queue.
XTPSRTNS	The RETAINS parameter specified the retention time after a successful transmission.

	XTPSRTNF	The RETAINF parameter specified the retention time after a failed transmission.
	XTPSRTYL	The RETYLF parameter specified the retry limit.
	XTPSRTYT	The RETYLT parameter specified the retry time.
XTPDSFLG	Flags containing data set information.	
	Flag	Meaning When Set to B'1'.
	XTPMDSCC	This data set contains carriage control characters.
	XTPDSJOB	This is the first data set in the job.
	XTPDSJNW	This data set is the JESNEWS data set.
	XTPDSEJB	This is the last data set in the job.
	XTPDSCON	This data set is part of a concatenation.
XTPPQLEN	Length of the print queue name in XTPPRTQU.	
XTPIALEN	Length of the value in XTPIPADR.	
XTPPRTQU	Name of the print queue, as specified in the PRTQUEUE JCL parameter, in a printer definition, or by the Routing exit. XTPPRTQU may not be available to the Response Notification exit if the Response code is X'08'.	
XTPIPADR	Host name or IP address of the target system, as specified in the DEST=IP JCL parameter, in a printer definition, or by the Routing exit; or the URL of the target system specified in the printer definition or the Routing exit. Field XTP_ADDR_TYPE identifies which value is in this field.	
XTPRETNS	Period of time IP PrintWay is to retain the data set on the JES spool after a successful transmission, as specified in the RETAINS JCL parameter or in a printer definition. IP PrintWay may have set a default value of zeros. See the note on page 95 for the format of this field. XTPRETNS may not be available to the Response Notification exit if the Response code is X'08'.	
XTPRETNF	Period of time IP PrintWay is to retain the data set on the JES spool after an all transmission retries has failed, as specified in the RETAINF JCL parameter or in a printer definition. IP PrintWay may have set a default value of zeros. See the note on page 95 for the format of this field. XTPRETNF may not be available to the Response Notification exit if the Response code is X'08'.	
XTPRTRYL	The number of times IP PrintWay is to retry transmission of the data set, as specified in the RETRYL JCL parameter or in a printer definition. IP PrintWay may have set a default value of zeros. XTPRTRYL may not be available to the Response Notification exit if the Response code is X'08'.	
XTPRTRYT	The period of time IP PrintWay is to wait between retries of the transmission, as specified in the RETRYT JCL parameter or in a printer definition. IP PrintWay may have set a default value of zeros. See the note on page 95 for the format of this field. XTPRTRYT may not be available to the Response Notification exit if the Response code is X'08'.	
XTPKEY	If XTP_ROUTING_KEY_TYPE is set to '1, contains the name of the	

printer definition, as specified in JCL, the printer definition, or by the Routing exit; the name is padded to the right with blanks.

XTPDEST	If XTP_ROUTING_KEY_TYPE is set to '0', contains the destination name of the data set, as specified in JCL, by the Routing exit, or defaulted by JES; the name is padded to the right with blanks. This field contains DFLTNTY if the DEST=IP parameter was specified in the OUTPUT JCL statement without the name of a printer definition. This field contains blanks if the DEST field in the printer definition is not specified.												
XTPCLASS	If XTP_ROUTING_KEY_TYPE is set to '0', contains the class of the data set, as specified in JCL, by the Routing exit, or defaulted by JES. This field contains blanks if the CLASS field in the printer definition is not specified.												
XTPFORMS	If XTP_ROUTING_KEY_TYPE is set to '0', contains the form name of the data set, as specified in JCL, by the Routing exit, or defaulted by JES; the name is padded to the right with blanks. This field contains blanks if the FORMS field in the printer definition is not specified.												
XTPRECFM	The record format, as indicated in the job file control block (JFCB). Refer to <i>OS/390 MVS Data Areas, Vol 3 (IVT-RCWK)</i> for a description of the JFCRECFM field in the JFCB.												
XTPMRECL	Maximum record length.												
XTPSWBTP	Address of the SWBTU for the data set. See “Scheduler Work Block Text Unit (SWBTU)” on page 93 for more information.												
XTPSWBTL	Length of the SWBTU area.												
XTPJSPAP	Address of the JES job separator page data area for the data set being processed (IAZJSPA). See “JES Job Separator Page Data Area (IAZJSPA)” on page 92 for more information.												
XTPCOPYS	Number of copies of the data set, as specified in JCL or defaulted by JES.												
XTPRSPCD	The Response Notification code. The response notification code is stored in XTPRSPCD. It may take any of the following values: <table> <tr> <th>Value</th><th>Meaning</th></tr> <tr> <td>00</td><td>PrintWay completed a successful transmission. After the successful retain time expires, the output will be purged from the JES spool.</td></tr> <tr> <td>04</td><td>The transmission was unsuccessful.</td></tr> <tr> <td>08</td><td>The initial processing of the data set by PrintWay was unsuccessful.</td></tr> <tr> <td>12</td><td>The transmission was unsuccessful. The retry limit is now 0. After the failure retain time expires, the output will be purged from the JES spool.</td></tr> <tr> <td>16</td><td>Because a spool I/O error occurred, the data set is unprintable.</td></tr> </table>	Value	Meaning	00	PrintWay completed a successful transmission. After the successful retain time expires, the output will be purged from the JES spool.	04	The transmission was unsuccessful.	08	The initial processing of the data set by PrintWay was unsuccessful.	12	The transmission was unsuccessful. The retry limit is now 0. After the failure retain time expires, the output will be purged from the JES spool.	16	Because a spool I/O error occurred, the data set is unprintable.
Value	Meaning												
00	PrintWay completed a successful transmission. After the successful retain time expires, the output will be purged from the JES spool.												
04	The transmission was unsuccessful.												
08	The initial processing of the data set by PrintWay was unsuccessful.												
12	The transmission was unsuccessful. The retry limit is now 0. After the failure retain time expires, the output will be purged from the JES spool.												
16	Because a spool I/O error occurred, the data set is unprintable.												
XTPIPORT	Port number on the target system, as specified in the PORTNO JCL parameter or in the printer definition.												

XTPOPTNS The transmission options. Refer to the ANFUEXTP macro for the field names. If you change a value in one of the fields, also set the length of the value in the length field that precedes that field.

IP PrintWay calls the same response exit for all data set transmissions. You can install different response exits for different PrintWay FSSs. See “Installing Exits” on page 124 for information.

Output from the Response Notification Exit

At return, restore the contents of all registers. Do not set a return code in register 15.

XTPMSGFL Flags indicating the destination of a message created by this exit. IP PrintWay sets these flags to B'0' before calling the exit. Set both flags to B'1' to issue the message to both destinations. Place the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.

Flag	Meaning When Set to B'1'.
------	---------------------------

XTPM2CON	Issue the message to the console.
-----------------	-----------------------------------

XTPM2MDS	Issue the message to the IP PrintWay message-log data set.
-----------------	--

XTPMSGLN Length of message, whose address is in XTPMSGP.

XTPMSGP Address of message created by this exit. IP PrintWay adds a message ID ANFM155I to the beginning of this message.

XTPRSPCD The Response Notification code.

Examples

1. To issue a message to the IP PrintWay message log when IP PrintWay has exhausted the transmission retry limit, code the Response user exit to:
 - Check that XTPRSPCD = 12.
 - Assemble the transmission failure message.
 - Place the address of the message into XTPMSGP in the ANFUEXTP control block to point to the message.
 - Place the length of the message in field XTPMSGLN.
 - Set XTPM2MDS to B'1' to indicate that the message should be printed to the log.
2. To issue a message to the operator's console when IP PrintWay has found a data set that failed pre-transmission processing:
 - Check that XTPRSPCD = 08.
 - Assemble the processing failure message.
 - Place the address of the message into XTPMSGP in the ANFUEXTP control block to point to the message.
 - Place the length of the message in field XTPMSGLN.
 - Set XTPM2CON to 1 to indicate that the message should be printed to the log.

Using the Block Letter Program (ANFUBLK)

The Begin Data Set exit and the End Data Set exit can use the Block Letter program (ANFUBLK) provided with IP PrintWay to build block letters suitable for

printing on separator pages. The block letters can be (1) straight or slanted, and (2) centered or left-justified. They are 12 characters high, using 12 printed lines, and can be either 8 or 12 characters wide.

ANFUBLK creates one line at a time for the exit to add to the data set. The exit must call ANFUBLK multiple times to receive all lines, until the ANFUBLK program indicates that this is the last line.

ANFUBLK produces block characters using tables in ANFUBTBL. The tables produce block characters from the following input. All other characters are translated to blanks.

- Uppercase letters: A - Z
- Symbols: \$, #, and @
- Numerals: 0 - 9
- Lowercase letters, a - z, which are translated to uppercase
- Superscript EBCDIC numerals, which are translated to numerals, 0 - 9

IP PrintWay provides the assembler language source code for ANFUBTBL in SYS1.SAMPLIB(ANFUBTBS). You can modify the source to create your own block letters. The comments in ANFUBTBS describe how to change the tables. After changing the tables, assemble ANFUBTBS, rename it to ANFUBTBL, and use the sample link job in SYS1.SAMPLIB(ANFULINK) to create your exit module.

Input to ANFUBLK

When calling the Block Letter program, set the registers as follows:

Register 1	Pointer to the address of the ANFUEXTP control block.
Register 13	Address of an 18-word save area for saving the caller's registers.
Register 14	Return address.
Register 15	Entry-point address.

Also, set the following fields in ANFUEXTP. The format of the ANFUEXTP control block is shown in Figure 26 on page 94.

XTPBIFLG	Flags indicating the type of block letters you want:
	Flag Meaning When Set to B'1'.
XTPBSLNT	Slant letters to the right.
XTPBPRFM	Create narrow letters for faster printing. Narrow letters are 8 characters wide, centered on an 80-character line; otherwise, letters are 12 characters wide, centered on a 132-character line.
XTPBLJST	Left justify the letters; otherwise, letters are centered.
	Note: During centering calculations, the Block Letter program ignores blanks to the right of the last character in field XTPBSTRG.
XTPBFRST	This is the first call to the Block Letter program to build block letters for the string whose address is in XTPBSTRG.
XTPBSTRG	String to convert to block letters, from one to eight characters, padded to the right with blanks.

Output from ANFUBLK

ANFUBLK sets register 15 to 0, and also sets the following fields in ANFUExTP:

XTPBLAST	Flag set to 1 if this is the last line. If this flag is set to 0, call the Block Letter program again.
XTPBLENG	Length of the line in field XTPBOUTP.
XTPBOUTP	The output line containing part of the block letter. The first character on each line contains a carriage control character.

Sample Exits

IP PrintWay provides the following sample exits written in assembler language in SYS1.SAMPLIB:

- ANFUXRTG — changes the IP address of the target system.
- ANFUXRT1 — changes the name of the components that contain transmission options.
- ANFUXRT2 — changes the IP address to a URL.
- ANFUXRT3 — changes the IP address and changes the print queue name to a port number.
- ANFUXRT4 — changes the printer definition name to DEST, CLASS, and FORMS values.
- ANFUXBD1 — adds a PCL5 printer setup string to the beginning of a data set.
- ANFUXBD2 — adds a separator page containing line data to the beginning of a data set. ANFUXBD2 uses the Block Letter program (ANFUBLK) to create block letters.
- ANFUXBD3 — requests printing a banner page before the first data set in a job but not before subsequent data sets in the job.
- ANFUXED1 — adds a separator page containing line data to the end of a data set. ANFUXED1 uses the Block Letter program (ANFUBLK) to create block letters.
- ANFUXRC1 — writes lines in a boustrophedon manner; that is, it writes alternate lines in opposite directions, from left to right and from right to left.
- ANFUXSMF — suppresses the SMF record for the JESNEWS data set.
- ANFUXMSG — suppresses messages ANFM700I and ANFM604I.
- ANFUXRSP — takes action according to the response notification code.

You can either view the samples online or print them. You can modify a sample exit, assemble, and install it to provide your own version of the exit. Object code for the sample exits is not provided.

Installing Exits

To install an exit, compile and link-edit the exit as follows:

- Name the exit:
 - For the Routing exit, use the name ANFUXRTG.
 - For the SMF exit, use the name ANFUXSMF.
 - For the Message exit, use the name ANFUXMSG.
 - For the Response Notification Exit, use the name ANFUXRSP.
 - For the Begin Data Set exit, the Record Exit, and the End Data Set exit, use any program name allowed by your system. Specify the exit name in one or more options entries in the printer definition.

- Link-edit the exit with the RENT and AMODE 31 attributes into an APF authorized library.
- If the Begin Data Set or End Data Set exit program calls the Block Letter program (ANFUBLK), use the sample JCL in SYS1.SAMPLIB(ANFULINK) to link-edit the exit with the Block Letter program (ANFUBLK) and the tables (ANFUBTBL).

Identify the library containing an exit in either:

- A STEPLIB statement in the IP PrintWay startup procedure.
You can provide a different exit program for each IP PrintWay FSS by using a unique STEPLIB for each startup procedure.
- A library concatenated to LNKLIST. If you place the exit in a LNKLIST library, then you must use the same exit program for every IP PrintWay startup procedure.

Modifying Exits

To modify any exit, replace the exit and then restart the IP PrintWay FSS.

Customizing PSF for OS/390 for Sharing Network Printers

PSF for OS/390 is a separately orderable product that runs on OS/390 and controls printing on AFP printers. When customizing PSF for OS/390 to send AFP data to IBM network printers, be sure to configure PSF for OS/390 for printer sharing if you also want to print on these printers from IP PrintWay. When configured for printer sharing, IP PrintWay can also send non-AFP data streams to the printer.

If you do not configure PSF for OS/390 for printer sharing, PSF for OS/390 does not release the printer to receive data from other sources.

Refer to *PSF for OS/390: Customization* for information about how to configure PSF for OS/390 for printer sharing.

Chapter 9. Installing and Customizing the Windows Client

Infoprint Server provides the following programs that run on Windows systems:

- OS/390 Printer Port Monitor for Windows 95/98, Windows NT, and Windows 2000
- AFP Printer Driver for Windows 95/98 and the AFP Printer Driver for Windows NT
- AFP Viewer plug-in for Windows 95/98 and Windows NT

See “Print Interface” on page 4 for an overview of the Windows client and how this component fits into your system.

Installing and Customizing the OS/390 Printer Port Monitor

The OS/390 Printer Port Monitor for Windows lets Windows users submit print jobs to Infoprint Server from Windows 95/98, Windows NT, and Windows 2000 systems using standard print-submission methods from any Windows application that supports printing.

Instead of using the OS/390 Printer Port Monitor for Windows, users can print using standard Windows Server Message Block (SMB) protocol, Internet Printing Protocol (IPP), or LPR to LPD protocol. To use these other printing protocols, users do not need to download and install the OS/390 Printer Port Monitor on their workstations. Table 5 compares the functions that the client can perform using these protocols.

Table 5. Comparison of Printing Methods

Function	Printer Port Monitor	SMB	IPP	LPR/LPD
Specify job attributes during printing	Yes	No	Yes ¹	No
Obtain status of print job	No	Yes ²	Yes	No

1. If the target printer is *not* IPP-enabled, only some IPP job attributes are supported. See “Customizing the IPP Workstation Client” on page 39 for information.
2. When the Windows job status window is open, OS/390 system performance might be degraded.

To use the OS/390 Printer Port Monitor for Windows, follow these steps:

1. Apply the following Infoprint Server PTFs on the OS/390 system:
 - PTF UW64005 (for APAR OW41376)
 - PTF for APAR OW41616
2. Customize the Printer Inventory Manager and the Print Interface components, including the Print Interface LPD.
3. Download and install the OS/390 Printer Port Monitor from the Printing Systems Company Web site (<http://www.ibm.com/printers>) or from one of the following OS/390 directories:
 - English version: `/usr/lpp/Printsrv/win/En_US`
 - Japanese version: `/usr/lpp/Printsrv/win/Ja_JP`
 - Spanish version: `/usr/lpp/Printsrv/win/Es_ES`

You can download the port monitor to a shared Windows folder, so that Windows users can install the port monitor from the shared folder. Visit the Web site or refer to *OS/390 Infoprint Server User's Guide* for complete information about how to download and install the port monitor.

4. Define an OS/390 printer to the Windows workstation using the Windows Add Printer Wizard. The Printer Inventory Manager daemon must be started before performing this task.

Refer to the Readme file for the OS/390 Printer Port Monitor for the steps required to add a printer and configure the Infoprint Server printer port. The following information is required:

- IP address or host name of the OS/390 system.
- The number of the port at which the Print Interface LPD is listening. Specify the same number as specified in the **lpd-port-number** statement in the **aopd.conf** file.
- The name of the printer definition for the OS/390 printer, which is defined in the Printer Inventory.
- The printer driver for the OS/390 printer. If the OS/390 printer is an IBM AFP printer, you can download the AFP printer driver from the OS/390 system or from the Web to the workstation. If the printer definition in the Printer Inventory is configured to transform PCL or PostScript data to AFP format, then you can use a generic PCL or PostScript printer driver on the workstation.

Installing the AFP Printer Driver

The AFP Printer Driver for Windows creates output files in AFP format for printing on AFP printers controlled by PSF.

To use the driver, download and install the driver from the Printing Systems Company Web site (<http://www.ibm.com/printers>) or from one of the following OS/390 directories:

- English version: **/usr/lpp/Printsrv/win/En_US**
- Japanese version: **/usr/lpp/Printsrv/win/Ja_JP**
- Spanish version: **/usr/lpp/Printsrv/win/Es_ES**

You can download the printer driver to a shared Windows folder, so that Windows users can install the driver from the shared folder. Visit the Web site or refer to *OS/390 Infoprint Server User's Guide* for complete information about how to download and install the viewer.

Installing the AFP Viewer Plug-In

The AFP Viewer plug-in for Windows lets Windows users view files that are in AFP format from a Web browser and print them to AFP and non-AFP printers.

To use the AFP Viewer plug-in, download and install the viewer from the the Printing Systems Company Web site, <http://www.ibm.com/printers>, or from the OS/390 directory, **/usr/lpp/Printsrv/win/En_US**. The Viewer plug-in is available only in English.

You can download the viewer to a shared Windows folder, so that Windows users can install the viewer from the shared folder.

Visit the Web site or refer to *OS/390 Infoprint Server User's Guide* for complete information about how to download and install the viewer. The Web site also contains information about how to customize font-mapping in the AFP Viewer plug-in.

Appendix A. Infoprint Server Files

This appendix describes Infoprint Server files that let you customize Infoprint Server.

aopd.conf

Purpose

This file contains statements that customize Infoprint Server.

Format

```
[# comment]
[ ascii-codepage = codepage ]
[ ebcdic-codepage = codepage ]
[ base-directory = path ]
[ inventory = inventory ]
[ ipp-port-number = portnumber ]
[ job-prefix = prefix ]
[ lpd-port-number = portnumber ]
[ snmp-community = name | public ]
[ start-daemons = { [ippd] [lpd] [snmpd] [xfd] }
```

ascii-codepage = *codepage*

The name of an ASCII code page. For code page names, refer to *OS/390 C/C++ Programming Guide*. Infoprint Server uses this code page in the following ways:

- By default, this code page appears in the **Printer code page** field in ISPF panels for new IP PrintWay printer definitions.
- Print Interface uses this code page as the source (document) code page when it translates documents received from remote systems, unless a document code page is specified in the printer definition or by the job submitter.
- Print Interface uses this code page as the target (printer) code page when it translates PostScript documents created with a non-ASCII code page.

Default: `ascii-codepage = IS08859-1`

base-directory = *pathname*

The directory path in which the Printer Inventory Manager creates Printer Inventory files. Also, Print Interface creates files in this directory.

Note: Do *not* change this statement while any Infoprint Server daemons are running.

Default: `base-directory = /var/Printsrv`

ebcdic-codepage = *codepage*

The name of an EBCDIC code page. For code page names, refer to *OS/390 C/C++ Programming Guide*. Infoprint Server uses this code page in the following ways:

- By default, this code page appears in the **Printer code page** field in ISPF panels for new PSF for OS/390 and General printer definitions.
- Print Interface uses this code page as the source (document) code page, when Print Interface cannot determine the user locale for the **lp** command.

Default: ebcdic-codepage = IBM-1047

inventory = *inventory*

The name assigned to the Printer Inventory. Specify 1–4 uppercase or lowercase letters and numbers; the name is case sensitive. Specify this name on the EXEC statements of the NetSpool and IP PrintWay startup procedures. Also, specify this name in the PSF for OS/390 startup procedure if you want PSF for OS/390 to use information in the Printer Inventory.

Note: Do *not* change this statement while any Infoprint Server daemons are running.

Default: inventory = AOP1

ipp-port-number = *portnumber*

The number of the port at which the IPP server waits for print requests. Port 631 is the well-known port for communication between IPP clients and IPP servers. If you specify a port other than 631, ensure that the port is not used by any other service on the OS/390 system.

Default: ipp-port-number = 631

job-prefix = *prefix*

A prefix added to the job identifier that Print Interface generates for data sets it allocates on the JES spool. This prefix lets the operator identify data sets on the JES spool allocated by Print Interface. Specify two letters, numbers, or national (@ \$ #) characters; the first character *cannot* be numeric.

The job identifier that Print Interface generates is not the same as the job identifier JES assigns, which is included in the fully-qualified data set name. This prefix is *not* added to the JES job identifier.

Default: job-prefix = PS

lpd-port-number = *portnumber*

The number of the port at which the Print Interface LPD waits for print requests. Port 515 is the well-known port for communication between LPRs and LPDs. If you specify a port other than 515, ensure that the port is not used by any other service on the OS/390 system and that you customize the LPR to send print requests to the new port. Windows users must specify this port when configuring the OS/390 Printer Port Monitor on a Windows system.

Default: lpd-port-number = 515

snmp-community = *name* | **public**

The name assigned to the SNMP community for making SNMP requests to the OS/390 system. This name must match (1) the community name provided to the OS/390 SNMP agent and (2) the community name defined to the SNMP manager (Network Printer Manager). If you do not provide a community name to the OS/390 SNMP agent, the OS/390 SNMP agent

accepts requests from any SNMP manager with the community name of **public**. Specify 1–32 uppercase or lowercase letters, numbers, or special characters. The community name is case sensitive. Note that the OS/390 agent and the SNMP manager may have other restrictions on the length of the community name or the allowed characters.

Default: `snmp-community = public`

start-daemons = { [ippd] [lpd] [snmpd] [xpd] }

The Infoprint Server daemons that are started when you issue the **aopstart** command. You can specify one or more of the following daemon names, enclosed in braces. The Printer Inventory Manager daemon always starts, regardless of which daemons you specify in this statement. To start only the Printer Inventory Manager daemon, type braces with no daemon names.

ippd	The aopippd daemon. This IPP server daemon processes print jobs submitted by an Internet Printing Protocol (IPP) client.
lpd	The aoplpd daemon. This line printer daemon processes print jobs submitted with commands, such as lpr , and by the OS/390 Printer Port Monitor for Windows.
snmpd	The aopsnmpd daemon. This SNMP subagent daemon provides status information to the OS/390 SNMP agent about printers controlled by PSF for OS/390.
xpd	The aopxpd daemon. This Transform Manager daemon manages the PCL and PostScript (and PDF) to AFP transforms provided by Infoprint Server Transforms for OS/390.

Examples: `start-daemons = {}`, `start-daemons = {lpd xpd}`

Default: `start-daemons = {lpd}`

Usage Notes

1. The Infoprint Server configuration file is an optional file. If present, it resides in the directory specified in the **AOPCONF** environment variable. If **AOPCONF** is not set, the default location is **/etc/Printsrv/aopd.conf**.
2. To create the configuration file, copy sample file **/usr/lpp/Printsrv/samples/aopd.conf** to **/etc/Printsrv/aopd.conf** or to the file location specified in **AOPCONF**.
3. All statements in this configuration file are optional. If you do not specify a statement, the default value is used.
4. If you edit the configuration file while Infoprint Server is running, you must stop and restart the Printer Inventory Manager daemon to pick up the changes. IBM recommends that you do *not* change the name of the Printer Inventory in the **inventory** statement after starting the Printer Inventory Manager daemon; if you do, you must also stop and restart any other components and products that use this name, including NetSpool, IP PrintWay, and PSF for OS/390.
5. When editing this file, follow these rules:
 - Use lowercase characters for the keywords and use uppercase or lowercase characters for the values, with blank characters before or after the equal sign, if desired.

- If a value contains blank characters or special characters (such as {}-> =,), enclose the value in single or double quotation marks.
- Start comments with: #
- Include blank lines, if desired.

Related Files

aopxfd.conf

Purpose

This file contains statements that customize the Infoprint Server Transform Manager. The Transform Manager manages the PCL to AFP and PostScript (and PDF) transforms provided by Infoprint Server Transforms for OS/390, a separate licensed program product.

To use the PCL and PostScript (and PDF) transforms, you must create an entry in this file for each transform. Infoprint Server Transforms also provides an SAP to AFP transform; however, to use the SAP to AFP transform, you do not create an entry in this file.

Format

```
# comment
transform transformname[_transformclass]
start-command = "daemon [ option ]..."
[ start-directory = path ]
[ maximum-idle-time = seconds ]
[ minimum-active = number ]
[ maximum-active = number ]
[ environment = { name -> value [ name -> value]... } ] ;
```

transform *transformname*[_*transformclass*]

This statement is required and must be the first statement in the entry. It identifies the start of a transform entry.

transformname

The name of the transform. To use transforms provided by Infoprint Server Transforms, specify one of the following names:

- **pcl2afp**, the name of the PCL to AFP transform
- **ps2afp**, the name of the PostScript (and PDF) to AFP transform

transformclass

The name of a transform class. This parameter does not apply for the PostScript (and PDF) to AFP transform; it is optional for the PCL to AFP transform.

Specify from 1 to 63 characters, including letters, numbers, or special characters. To use this transform class, the administrator names the class in the **-c** option of the **pcl2afp.dll** filter in the printer definition in the Printer Inventory. Also, a user can name the class in the **-c** option of the **pcl2afp** command, and in **filter-options** attribute of the **lp** command and AOPPRINT JCL procedure.

Default: If this parameter is omitted, this entry is used only if the filter or command does not name a class.

Example: transform pcl2afp_letter_300

To use this transform class on the **pcl2afp** command, type:

```
pcl2afp -c letter_300
```

Default: None

start-command = "daemon [option]... "

The name of the transform daemon and any options accepted by the transform daemon. This statement is required. To use a transform daemon provided by Infoprint Server Transforms, specify one of the following values:

- **pcl2afpd**, the PCL to AFP transform daemon
- **ps2afpd**, the PostScript (and PDF) to AFP transform daemon

If the transform daemon is not in a directory identified in the PATH environment variable, then specify the full directory path name of the daemon.

The **pcl2afpd** daemon supports the following option:

-m nnn{K | M}

The number of bytes of memory the transform daemon can use to perform transforms. Specify the number of bytes in either kilobytes or megabytes. The amount required depends on the compression type, the image being compressed, fonts, and so on.

Use the following algorithm or refer to the following table to determine the amount to initially specify; if the transform fails due to lack of memory, then specify a larger amount of storage.

$((\text{width in pixels} / 8) * \text{height in pixels}) * 3$

Width in pixels = (width in inches * resolution)

Height in pixels = (height in inches * resolution)

The following table shows the amount of storage for different paper sizes and resolutions (240, 300, and 600 pixels per inch), using the previous algorithm:

Paper Size	240	300	600
Letter	2.0M	3.2M	12.6M
Legal	2.6M	4.0M	16.1M
A4	2.1M	3.3M	13.1M

Example: start-command = "pcl2afpd -m 4M"

Default: 6M

Note: If you specify this option, enclose the keyword and value in single or double quotation marks.

Default: None

start-directory = path

Path where the fonts and resources used by the PostScript (and PDF) to AFP transform are located. This statement does not apply for the PCL to AFP transform; it is optional for the PostScript (and PDF) to AFP transform. If you installed Infoprint Server Transforms fonts and resources in the default directory, specify **/usr/lpp/Printsrv/ps2afp**.

Default: Current directory

maximum-idle-time = seconds

The number of seconds before the Transform Manager shuts down an idle transform daemon and system resources are freed. This statement is optional; however, to avoid having many transform daemons active, consider specifying either this statement or the **maximum-active statement** (or both). The number you specify must be greater than 0.

Default: An idle transform daemon is not shut down.

minimum-active = number

The number of transform daemons that the Transform Manager starts, and also the minimum number of transform daemons that the Transform Manager keeps active (that is, not shut down), even when the maximum-idle-time expires for an idle transform daemon. The number you specify must be greater than 0. This statement is optional.

Default: minimum-active = 0

maximum-active = number

The maximum number of transform daemons that the Transform Manager keeps active. When this number is reached, the Transform Manager does not start a new transform daemon to perform the transform; therefore, the transform request waits until a transform daemon is available. This statement is optional; however, to avoid having many transform daemons active, consider specifying either **maximum-active** or **maximum-idle-time**, or both. The number you specify must be greater than 0 and greater than or equal to the number specified in **minimum-active**.

Default: No maximum number; transform daemons are started when needed.

environment = {name -> value [name -> value]... }

Environment variables that define the transform environment for PCL to AFP transforms. Enclose the environment variables in braces. The values in these environment variables override values of environment variables with the same names that were set when the **aopstart** command was issued. Following are the environment variables you can specify:

Environment Variable	Meaning
AOP_PAGE_HEIGHT	Page height in inches (i) or millimeters (m). Default: AOP_PAGE_HEIGHT -> 11i
AOP_PAGE_WIDTH	Page width in inches (i) or millimeters (m). Default: AOP_PAGE_WIDTH -> 8.5i
AOP_RESOLUTION	Output resolution in pixels per inch. Default: AOP_RESOLUTION -> 240
AOP_HORIZONTAL_MARGINS	Left and right margins in inches (i) or millimeters (m). Default: AOP_HORIZONTAL_MARGINS -> 0.167i
AOP_VERTICAL_MARGINS	Top and bottom margins in inches (i) or millimeters (m). Default: AOP_VERTICAL_MARGINS -> 0.167i

Usage Notes

1. This Transform configuration file is required if you start the Transform Manager daemon. Some statements in the file are optional; if you do not specify an optional statement, the default value is used.
2. The Transform configuration file resides in the directory specified in the **AOPXFD_CONF** environment variable. If this environment variable is not set, the default location is **/etc/Printsrv/aopxfd.conf**.
3. To create the Transform configuration file, copy sample file **/usr/lpp/Printsrv/samples/aopxfd.conf** to **/etc/Printsrv/aopxfd.conf** or to the directory specified in **AOPXFD_CONF**.
4. If you edit the configuration file after starting the transform daemon, use the **aopstop -d xfd** command to stop the transform daemon, and use the **aopstart** command to restart it.
5. When editing this file, follow these rules:
 - End the last statement in a transform entry with a semicolon.
 - Use lowercase characters for the keywords and uppercase or lowercase characters for values, with blank characters before or after the equal sign, if desired.
 - Use uppercase characters for the environment variables.
 - If a value contains blank characters or special characters (such as {}-> =,), enclose the value in single or double quotation marks.
 - Start comments with: #
 - Include blank lines if desired.

Examples

```
transform ps2afp
    start-command = ps2afpd
    start-directory = /usr/lpp/Printsrv/ps2afp
    minimum-active = 1
    maximum-active = 2
    maximum-idle-time = 300    # 5 minutes
;

transform pcl2afp
    start-command = "pcl2afpd -m 4M"
    minimum-active = 1
    maximum-active = 2
    maximum-idle-time = 300    # 5 minutes
    environment = {
        AOP_PAGE_HEIGHT -> 11i
        AOP_PAGE_WIDTH -> 8.5i
        AOP_RESOLUTION -> 240
        AOP_HORIZONTAL_MARGINS -> 0.167i
        AOP_VERTICAL_MARGINS -> 0.167i
    }
;

transform pcl2afp_letter_300
    start-command = pcl2afpd
    minimum-active = 1
    maximum-active = 2
    maximum-idle-time = 300    # 5 minutes
    environment = {
        AOP_PAGE_HEIGHT -> 11i
        AOP_PAGE_WIDTH -> 8.5i
        AOP_RESOLUTION -> 300
        AOP_HORIZONTAL_MARGINS -> 0.167i
        AOP_VERTICAL_MARGINS -> 0.167i
    }
;
```

Related Files

`aopd.conf`

Appendix B. Environment Variables

This appendix lists environment variables that affect processing of Infoprint Server.

AOPCONF The full path name of the Infoprint Server configuration file. This environment variable is optional; if you created the configuration file in `/etc/Printsrv/aopd.conf`, you do not need to set this environment variable.

AOPTIONS

Specifies a string of job attributes and values that are to be in effect for each **lp** command. The **lp** command includes the attributes specified in this environment variable before the attributes specified, if any, on the **-o** option of the **lp** command. Because the **lp** command reads the value of the AOPTIONS environment variable before the options you specify on the command line, a user can override the values of this variable. This variable is *optional*; use it to specify job attributes that are constant for all print jobs. Refer to *OS/390 Infoprint Server User's Guide* for the format of job attributes and values.

AOPPATH Path where job attribute files, used by the **lp** command and AOPPRINT procedure, are located. This environment variable is *optional*; if you have not created attribute files for the **lp** command, you do not need to set this environment variable. Refer to *OS/390 Infoprint Server User's Guide* for information about attribute files.

AOP_SAP2AFP_RESOURCES

Path where resources used by the SAP to AFP transform are located. This variable is *optional*. If you installed Infoprint Server Transforms resources in the default directory, `/usr/lpp/Printsrv/sap2afp/`, you do not need to set this environment variable.

AOPXFD_CONF

The full path name of the transform configuration file. This variable is *optional*; if you created the configuration file in `/etc/Printsrv/aopxfd.conf`, you do not need to set this environment variable.

CLASSPATH The full path names of Infoprint Server IPP files. If you installed Infoprint Server IPP files in the default directory, `/usr/lpp/Printsrv`, you do not need to modify this environment variable. If you installed Infoprint Server files in a different directory, add the following values to any existing values in this variable:

- `directory/classes/ipp.jar`
- `directory/classes/ippserver.jar`
- `directory/classes/ippclient.jar`
- `directory/classes/ippreal.jar`

directory is the directory that contains the files.

JAVA_HOME The path used to locate Java files. This environment variable is *optional*; if you installed Java files in the default directory, `/usr/lpp/java/J1.1`, you do not need to set this environment variable.

LIBPATH	The path used to locate dynamic link library (DLL) files. This environment variable is <i>required</i> . If you installed Infoprint Server files in the default directory, add /usr/lpp/Printsrv/lib to any existing values. If you installed Infoprint Server files in a different directory, add the directory to any existing values.						
MANPATH	<p>The path of directories that contain the man pages. This environment variable is <i>required</i>. If the LANG environment variable identifies the language in which you want to view Infoprint Server man pages, add /usr/lpp/Printsrv/man/%L to the values in this variable; otherwise, add one of the following values to any existing values:</p> <table> <tr> <td>English</td><td>/usr/lpp/Printsrv/man/En_US</td></tr> <tr> <td>Japanese</td><td>/usr/lpp/Printsrv/man/Ja_JP</td></tr> <tr> <td>Spanish</td><td>/usr/lpp/Printsrv/man/Es_ES</td></tr> </table> <p>Note: Add the new directory path <i>before</i> /usr/man/%L in the MANPATH environment variable, so that the Infoprint Server versions of the lp, lpstat, and cancel man pages are displayed.</p>	English	/usr/lpp/Printsrv/man/En_US	Japanese	/usr/lpp/Printsrv/man/Ja_JP	Spanish	/usr/lpp/Printsrv/man/Es_ES
English	/usr/lpp/Printsrv/man/En_US						
Japanese	/usr/lpp/Printsrv/man/Ja_JP						
Spanish	/usr/lpp/Printsrv/man/Es_ES						
NLSPATH	<p>The path of directories that contain message catalogs. This environment variable is <i>required</i>. If the LANG environment variable identifies the language in which you want to receive Infoprint Server messages, add /usr/lpp/Printsrv/%L/%N to the values in this variable; otherwise, add one of the following values:</p> <table> <tr> <td>English</td><td>/usr/lpp/Printsrv/En_US/%N</td></tr> <tr> <td>Japanese</td><td>/usr/lpp/Printsrv/Ja_JP/%N</td></tr> <tr> <td>Spanish</td><td>/usr/lpp/Printsrv/Es_ES/%N</td></tr> </table> <p>%L represents the value of the LANG environment variable. %N is the catalog filename.</p>	English	/usr/lpp/Printsrv/En_US/%N	Japanese	/usr/lpp/Printsrv/Ja_JP/%N	Spanish	/usr/lpp/Printsrv/Es_ES/%N
English	/usr/lpp/Printsrv/En_US/%N						
Japanese	/usr/lpp/Printsrv/Ja_JP/%N						
Spanish	/usr/lpp/Printsrv/Es_ES/%N						
PATH	The path used to locate executables. This environment variable is <i>required</i> . If you installed Infoprint Server executables in the default directory, add /usr/lpp/Printsrv/bin to the values in this variable. Be sure to add the directory <i>before</i> /bin in the PATH environment variable to ensure that the Infoprint Server versions of the lp , lpstat , and cancel commands are invoked.						
PRINTER or LPDEST	<p>The default printer for the lp command. The value in LPDEST overrides the value in PRINTER.</p> <p>This variable is <i>optional</i>. You can use Infoprint Server ISPF panels to define a default printer; see <i>OS/390 Infoprint Server Operation and Administration</i> for information. The printer named in either LPDEST or PRINTER overrides the printer named on the ISPF configuration panel.</p>						

Appendix C. SCS Printer Simulation

NetSpool transforms the SNA character stream (SCS) for an LU type 1 printer into a VBA (variable blocked with ASA carriage control) data set. The following table describes the code points that are undefined, unsupported, supported with defaults or fully supported.

NetSpool returns the following SNA sense codes for errors found in the SCS data stream:

- SNA sense code of function error (X'10030000') for undefined and unsupported code points.
- SNA sense code of parameter error (X'10050000') for supported code points with invalid parameters or without all parameters available in the same chain.
- SNA sense code of data error (X'10010000') for invalid characters in a DBCS string.

Table 6. SCS Code Points

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	Null	(1)	(1)	(1)	SEL (9)	HT	RNL (4)	(1)	GE (3)	SPS (3)	RPT (3)	VT	FF	CR	SO (11)	SI (11)
1x	(1)	DC1 (3)	DC2 (3)	DC3 (3)	ENP (3)	NL	BS	POC (3)	(1)	(1)	UBS (3)	CU1 (3)	IFS (4)	IGS (4)	IRS (4)	IUS (7)
2x	(1)	(1)	(1)	WUS	INP (3)	LF			SA (12)	(1)	SW (3)	CSP (10)	(1)	(1)	(1)	BEL (3)
3x	(1)	(1)	SYN (3)	IR (4)	PP	TRN	EBS (6)	(1)	SBS (3)	IT (3)	RFF (5)	CU3 (3)	DC4 (3)	(1)	(1)	SUB
4x		RSP (7)														
5x																
6x																
7x																
8x																
9x																
Ax																
Bx																
Cx											SHY (8)					
Dx																
Ex		ESP (7)														
Fx																

Notes on the table:

1. Undefined code point - function error.
2. Unsupported code point - function error.
3. Defaults to no operation - function ignored.

4. Defaults to new line (NL).
5. Defaults to form feed (FF).
6. Defaults to backspace (BS).
7. Defaults to space (X'40').
8. Defaults to dash (X'60').
9. Vertical channel select is supported. Select left/right platten is ignored. Select magnetic stripe reader/writer is unsupported.
10. Set Horizontal Format and Set Vertical Format are supported. Start of Format is ignored if at left margin and defaults to new line (NL) if not at left margin. Set Line Density, Set Graphic Escape Action, Set Chain Image and Set Print Density are ignored.
11. Shift Out (SO) indicates the start of a string of double-byte character set (DBCS) data. Shift In (SI) indicates the end. Valid characters in the DBCS string are X'4040' and any pair of bytes, each in the range X'41' to X'FE'. NetSpool rejects invalid characters and DBCS strings that do not complete in the same chain with an SNA sense code of data error (X'10010000').
12. Set Attribute (SA) X'2843F8' indicates the start of double-byte character set (DBCS) data. Set Attribute X'284300', X'280000', or end-of-chain indicates the end. Valid characters in the DBCS string are X'4040' and any pair of bytes, each in the range X'41' to X'FE'. NetSpool rejects invalid characters with an SNA sense code of data error (X'10010000').
 NetSpool converts SA code points that indicate the start and end of a DBCS string into Shift Out (SO) and Shift In (SI) line-data controls. NetSpool converts valid SCS code points in the DBCS string to the appropriate line-data controls, delimited by SI and SO line-data controls.
 NetSpool ignores Set Attribute code points with attributes of Color, Highlighting, or Field Outlining.

NetSpool passes all unspecified code points to JES as EBCDIC data.

Refer to *SNA - Sessions Between Logical Units* for more information about SCS data streams. Refer to *IBM 3270 Kanji Data Streams* for more information about DBCS SCS data streams.

Appendix D. 3270 Data Streams

NetSpool transforms the 3270 data streams for an LU type 0 or LU type 3 printer into a VBA (variable blocked with ASA carriage control) data set. The following tables describe the code points that are supported for 3270 data streams.

Command Codes

Only one command is allowed in each RU chain. The command must be the first byte of the RU chain.

COMMAND	EBCDIC	COMMENTS
W	X'F1'	Write
EW	X'F5'	Erase/Write
EWA	X'7E'	Erase/Write Alternate
EAU	X'6F'	Erase All Unprotected
Other command codes		Function not supported. Return SNA sense code X'1003000'

Control Codes

The control codes have an EBCDIC value in the range of hexadecimal 00 (X'00') through hexadecimal 3F (X'3F').

CONTROL CODE	EBCDIC	COMMENTS
NL	X'15'	New Line
EM	X'19'	End of Message
FF	X'0C'	Forms Feed
CR	X'0D'	Carriage Return
SF	X'1D'	Start Field
SBA	X'11'	Set Buffer Address
IC	X'13'	Insert Cursor
PT	X'05'	Program Tab
RA	X'3C'	Repeat to Address
SFE	X'29'	Start Field Extended (See attributes)
SA	X'28'	Set Attribute (See attributes)
MF	X'2C'	Modify Field (See attributes)
EUA	X'12'	Erase Unprotected to Address
GE	X'08'	Graphic Escape (Defaults to no operation-function, ignored and removed from data stream.)
SO	X'0E'	Shift Out
SI	X'0F'	Shift In
SYN	X'32'	SYN Character (Defaults to no operation-function, ignored and removed from data stream.)

Other control codes		Function not supported. Return SNA sense code X'1003000'
---------------------	--	--

Attribute Types

ATTRIBUTE TYPE	EBCDIC	SFE, MF Orders	SA Order	COMMENTS
Character Attribute Reset	X'00'	N/A	X	Set character set attribute to single byte character set (default)
Character Set	X'43'	X	X	When attribute value is X'00'-X'7F', character set attribute is set to single byte. When attribute value is X'00'-X'7F', character set attribute is set to double byte.
3270 Field Attribute	X'C0'	X	N/A	Field attribute bit definitions supported: <ul style="list-style-type: none"> • Bit 2 = B'0' Field is unprotected. • Bit 2 = B'1' Field is protected. • Bits 4,5 = B'11' Field is nonprintable. • Bits 4,5 Other settings are ignored. All other bit definitions are ignored.
Other Valid Attributes	X'41', X'42', X'45', X'46', X'C2'	X	X	Tolerated; defaults to no operation. Attributes are ignored.
Invalid Attributes		X	X	Function not supported. Return SNA sense code X'1003000'.

Note:

N/A The attribute type does not apply to the order.

X The attribute type does apply to the order.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Mail Drop 001W
Boulder, CO 80301
U.S.A

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Programming Interfaces

This publication documents programming interfaces intended to allow you to write programs to obtain the services of Infoprint Server.

Trademarks

The following terms appear in this publication and are either trademarks or registered trademarks of International Business Machines Corporation in the United States, in other countries, or both:

AFP
AIX
Application System/400
AS/400
C/370
CICS
eNetwork
IBM
IMS
IMS/ESA
Infoprint
Intelligent Printer Data Stream
IP PrintWay
IPDS
Language Environment
MVS/ESA
NetSpool
OpenEdition
Operating System/2
OS/2
OS/390
Print Services Facility
PrintWay
RACF
RETAIN
RS/6000
S/370
S/390
SecureWay

VTAM
WIN-OS/2

The following terms appear in this publication and are either trademarks or registered trademarks of subsidiaries of International Business Machines Corporation in the United States, in other countries, or both:

- Freelance is a trademark of Lotus Development Corporation. Lotus is a registered trademark of Lotus Development Corporation.

The following terms appear in this publication and are either trademarks or registered trademarks of other companies in the United States, in other countries, or both:

- Java is a registered trademark of Sun Microsystems, Inc.
- MCS is a registered trademark of Intel.
- Microsoft, Windows, and Windows NT are registered trademarks of Microsoft Corporation.
- UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names may be trademarks or service marks of others.

Glossary

Sources

This glossary defines technical terms and abbreviations used in Infoprint Server documentation. If you do not find the term you are looking for, see the index of this publication or view *IBM Dictionary of Computing*, located at: <http://www.ibm.com/networking/nsg/nsgmain.htm>

Definitions reprinted from *American National Dictionary for Information Processing Systems* are identified by the symbol (A) following the definition.

Definitions reprinted from a published section of the International Organization for Standardization (ISO) *Vocabulary—Information Processing* or from a published section of *Vocabulary—Office Machines* developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Committee (ISO/IEC JTC1/SC1) are identified by the symbol (I) following the definition. Because many ISO definitions are also reproduced in *American National Dictionary for Information Processing Systems*, ISO definitions may also be identified by the symbol (A).

Definitions reprinted from working documents, draft proposals, or draft international standards of ISO Technical Committee 97, Subcommittee 1 (Vocabulary), Joint Technical Committee 1 are identified by the symbol (T) following the definition, indicating that final agreement has not yet been reached among its participating members.

This publication contains terms and terms and definitions from *Information Technology—Portable Operating System Interface (POSIX)* from the POSIX series of standards for applications and user interfaces to open systems, copyrighted by the Institute of Electrical and Electronics Engineers (IEEE). Copies of all POSIX drafts and standards may be purchased from IEEE at 1-800-678-IEEE.

- Definitions identified by (POSIX.0) are from *Part 0: Standards Project, Draft Guide to the POSIX Open System Environment* P1003.0 Draft 15 (June 1992), an unapproved draft subject to change.

- Definitions identified by (POSIX.1) are from *Part 1: System Application Program Interface (API) [C Language]* approved September 28, 1990, as IEEE Std 1003.1-1990 by the IEEE Standards Board, and adopted in 1990 as an International Standard (ISO/IEC 9945-1: 1990) by the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC).
- Definitions identified by (POSIX.2) are from *Part 2: Shell and Utilities* P1003.2.

Definitions that are specific to IBM products are so labeled; for example, “In TCP/IP,” or “In Infoprint Server.”

References

The following cross-references are used in this glossary:

Contrast with. This refers to a term that has an opposite or substantively different meaning.

See. This refers to multiple-word terms in which this term appears.

See also. This refers to related terms that have similar, but not synonymous, meanings.

Synonym for. This appears in the commentary of a less desirable or less specific term and identifies the preferred term that has the same meaning.

Synonymous with. This appears in the commentary of a preferred term and identifies less desirable or less specific terms that have the same meaning.

Numerics

3270 data stream. Data transferred from or to an allocated primary or tertiary device, or to the host system, as a continuous stream of data and 3270 Information Display System control elements in character form.

A

abend. Termination of a task before its completion because of an error condition that cannot be resolved by recovery facilities while the task is executing.

ACB. Access method control block.

access method control block (ACB). A control block that links an application program to VTAM.

ACIF. (1) AFP conversion and indexing facility. (2) A PSF utility program that converts a print file into AFP, MO:DCA-P, creates an index file for input data, and collects resources used by an AFP document into a separate file.

Advanced Function Presentation (AFP). A set of licensed programs, together with user applications, that use the all-points-addressable concept to print on presentation devices. AFP includes creating, formatting, archiving, retrieving, viewing, distributing, and printing information.

AFP. Advanced Function Presentation.

AFP Printer Driver for Windows. A component of Infoprint Server for OS/390 that runs on a Windows 95 or Windows NT workstation and creates output in AFP format, for printing on AFP printers.

AFP Viewer plug-in for Windows. A component of Infoprint Server for OS/390 that runs on a Windows 95 or Windows NT workstation and allows you to view files in AFP format.

AIX operating system. IBM's implementation of the UNIX operating system. The RS/6000[®] system, among others, runs the AIX operating system.

alphanumeric character. A letter or a number.

ASCII (American Standard Code for Information Interchange). The standard code, using a coded character set consisting of 7-bit coded characters (8-bit including parity check), that is used for information interchange among data processing systems, data communication systems, and associated equipment. The ASCII set consists of control characters and graphic characters. (A)

Note: IBM has defined an extension to ASCII code (characters 128–255).

B

banner page. A page printed before the data set is printed.

binary data. (1) Any data not intended for direct human reading. Binary data may contain unprintable characters, outside the range of text characters. (2) A type of data consisting of numeric values stored in bit patterns of 0s and 1s. Binary data can cause a large number to be placed in a smaller space of storage.

BIND. In SNA, a request to activate a session between two logical units (LUs).

broadcast. (1) Transmission of the same data to all destinations. (T) (2) Simultaneous transmission of data to more than one destination.

buffer. A portion of storage used to hold input or output data temporarily.

burst. To separate continuous-forms paper into single sheets.

C

carriage control character. An optional character in an input data record that specifies a write, space, or skip operation.

carriage return (CR). (1) A keystroke generally indicating the end of a command line. (2) In text data, the action that indicates to continue printing at the left margin of the next line. (3) A character that will cause printing to start at the beginning of the same physical line in which the carriage return occurred.

case-sensitive. Pertaining to the ability to distinguish between uppercase and lowercase letters.

catalog. (1) A directory of files and libraries, with reference to their locations. (2) To enter information about a file or a library into a catalog. (3) The collection of all data set indexes that are used by the control program to locate a volume containing a specific data set.

CICS. Customer Information Control System.

client. A functional unit that receives shared services from a server. See also *client-server*.

client-server. In TCP/IP, the model of interaction in distributed data processing in which a program at one site sends a request to a program at another site and awaits a response. The requesting program is called a client; the answering program is called a server.

code page. (1) A table showing codes assigned to character sets. (2) An assignment of graphic characters and control function meanings to all code points. (3) Arrays of code points representing characters that establish ordinal sequence (numeric order) of characters. (4) A particular assignment of hexadecimal identifiers to graphic elements.

code point. A 1-byte code representing one of 256 potential characters.

coexistence. Two or more systems at different levels (for example, software, service or operational levels) that share resources. Coexistence includes the ability of a system to respond in the following ways to a new function that was introduced on another system with which it shares resources: ignore a new function, terminate gracefully, support a new function.

connection. In TCP/IP, the path between two protocol applications that provides reliable data stream delivery service. In Internet communications, a connection extends from a TCP application on one system to a TCP application on another system.

copy group. One or more copies of a page of paper. Each copy can have modifications, such as text suppression, page position, forms flash, and overlays.

Customer Information Control System (CICS). An IBM licensed program that enables transactions entered at remote terminals to be processed concurrently by user-written application programs. It includes facilities for building, using, and maintaining databases.

D

daemon. A program that runs unattended to perform a standard service. Some daemons are triggered automatically to perform their task; others operate periodically.

data set. The major unit of data storage and retrieval, consisting of a collection of data in one of several prescribed arrangements and described by control information to which the system has access.

data stream. (1) All information (data and control commands) sent over a data link usually in a single read or write operation. (2) A continuous stream of data elements being transmitted, or intended for transmission, in character or binary-digit form, using a defined format.

DBCS. Double-byte character set.

default. A value, attribute, or option that is assumed when no alternative is specified by the user.

directory. (1) A type of file containing the names and controlling information for other files or other directories. Directories can also contain subdirectories, which can contain subdirectories of their own. (2) A file that contains directory entries. No two directory entries in the same directory can have the same name. (POSIX.1). (3) A file that points to files and to other directories. (4) An index used by a control program to locate blocks of data that are stored in separate areas of a data set in direct access storage.

DLL filter. A filter that provides one or more of these functions in a dynamic load library - `init()`, `prolog()`, `process()`, `epilog()`, and `term()`. See **cfilter.h** and **cfilter.c** in the `/usr/lpp/Printsrv/samples/` directory for more information. See also **filter**. Contrast with **DLL filter**.

dotted decimal notation. The syntactical representation for a 32-bit integer that consists of four 8-bit numbers written in base 10 with periods (dots) separating them. It is used to represent IP addresses.

double-byte character set (DBCS). A set of characters in which each character is represented by a two-bytes code. Languages such as Japanese, Chinese, and Korean, which contain more symbols than can be represented by 256 code points, require double-byte character sets. Because each character requires two bytes, the typing, display, and printing of DBCS characters requires hardware and programs that support DBCS. Contrast with *single-byte character set*.

download. To transfer data from one computer for use on another one. Typically, users download from a larger computer to a diskette or fixed disk on a smaller computer or from a system unit to an adapter.

Download for OS/390. A feature of PSF for OS/390 that allows the PSF program to automatically send data sets from the JES spool, without formatting them, directly to either Infoprint Manager for AIX or EDMSuite OnDemand, using the TCP/IP protocol. Infoprint Manager for AIX and EDMSuite OnDemand servers receive the data sets into files, which can be automatically formatted and printed by Infoprint Manager for AIX or loaded into EDMSuite OnDemand.

drain. An operator action to halt the flow of jobs to a printer, usually to stop the printer or to change print options.

E

EBCDIC. Extended binary-coded decimal interchange code. A coded character set consisting of 8-bit coded characters. (A)

encryption. In computer security, the process of transforming data into an unintelligible form in such a way that the original data either cannot be obtained or can be obtained only by using a decryption process.

environment variable. (1) A name associated with a string of characters, made available to the programs that you run. (2) A variable that describes the operating environment of the process and typically includes information about the home directory, command search path, the terminal in use, and the current time zone. (3) A variable included in the current software environment that is available to any called program that requests it.

F

file. (1) A set of related records treated as a unit. (2) A collection of related data that is stored and retrieved by an assigned name. (3) Linear data that can be opened, written, read, and closed. A file can also contain information about the file, such as authorization information. The name used to obtain a file includes the directories in the path to the file. (4) Strings of characters with no additional structure. Structure is assumed only by the processing programs. Files can be

located relative to the current directory or by an absolute pathname. (5) An object that can be written to, or read from, or both. A file has certain attributes, including access permissions and type. File types include regular file, character special file, block special file, FIFO special file, and directory. Other types of files may be defined by the implementation. (POSIX.1) In the OS/390 UNIX System Services implementation, the file system does not support block special files, but it does support symbolic link files. (6) A collection of information or data that is organized by some method (relative, indexed, or serial, for example) and stored on a device such as a disk.

file system. (1) A collection of files and directories. (2) The collection of files and file management structures on a physical or logical mass storage device, such as a disk or disk partition. A single device can contain several file systems. (3) A mountable subtree of the directory hierarchy. (4) A collection of files and certain of their attributes. A file system provides a name space for file serial numbers referring to those files. (POSIX.1).

filter. In Print Interface, a program that can add, delete, or modify input data before Print Interface writes the data to the JES spool. Print Interface provides support for two types of filter programs: DLL filters and UNIX filters. See also DLL filter and UNIX filter.

font. (1) A family or assortment of characters of a given size and style; for example, 9 point Bodoni Modern. (A) (2) One size and one typeface in a particular type family, including letters, numerals, punctuation marks, special characters, and ligatures. (3) A paired character set and code page that can be used together for printing a string of text characters. A double-byte font can consist of multiple pairs of character sets and code pages.

form definition. A resource used by PSF that defines the characteristics of the form, which includes such functions as overlays to be used (if any), paper source (for cut-sheet printers), duplex printing, text suppression, the position of MO:DCA-P data on the form, and the number and modifications of a page.

FSA. Functional subsystem application.

FSS. Functional subsystem.

functional subsystem (FSS). An address space uniquely identified as performing a specific function related to the JES.

functional subsystem application (FSA). The functional application program managed by the functional subsystem.

H

hexadecimal. (1) Pertaining to a selection, choice, or condition that has 16 possible different values or states. (I) (2) Pertaining to a fixed-radix numeration system, with radix of 16. (I) (3) Pertaining to a system of numbers to the base 16; hexadecimal digits range from 0 through 9 and A through F, where A represents 10 and F represents 15.

hiperspace. The space used for paging by the OS/390 operating system.

HFS data set. A hierarchical file system data set, which is used to store, and is essentially identified with, a file system.

home directory. (1) The current directory associated with the user at the time of login. (POSIX.2) (2) A directory associated with an individual user. (3) The user's current directory on login or after issuing the **cd** command with no argument.

host. In the Internet suite of protocols, an end system. The end system can be any system; it does not have to be a mainframe.

host address. See *IP address*.

host name. In the Internet suite of protocols, the name given to a machine. Sometimes, "host name" is used to mean *fully qualified domain name*; other times, it is used to mean the most specific subname of a fully qualified domain name. For example, if *boulder.vnet.ibm.com* is the fully qualified domain name, either of the following may be considered the host name:

- boulder.vnet.ibm.com
- boulder

I

IMS. Information Management System

Infoprint Server for OS/390. An element of OS/390 V2R8 and higher that supports printing on OS/390 printers, including local printers and remote printers in a TCP/IP network. Infoprint Server lets users submit print requests from remote workstations in a TCP/IP network, from OS/390 UNIX System Services applications, from batch applications, and from VTAM applications, such as CICS or IMS applications. Infoprint Server consists of the following components:

- IP PrintWay
- NetSpool
- Print Interface
- Printer Inventory Manager
- Transform Manager and Infoprint Server Transforms for OS/390
- SNMP subagent
- Windows client
 - IBM AFP Printer Driver for Windows

- IBM AFP Viewer plug-in for Windows
- OS/390 Printer Port Monitor for Windows

Information Management System (IMS). A database/data communication system that can manage complex databases and networks.

inline resource. A resource contained in the print data set.

Internet. A wide area network connecting thousands of disparate networks in industry, education, government, and research. The Internet network uses TCP/IP as the protocol for transmitting information.

Internet Printing Protocol (IPP). An application-level protocol that enables distributed printing on the Internet. IPP uses a client/server architecture and defines the interactions between IPP clients (typically work-stations) and IPP servers.

Internet Protocol (IP). A protocol used to route data from its source to its destination in an Internet environment.

IP. Internet Protocol.

IP address. (1) In the Internet suite of protocols, the 32-bit address of a machine, expressed in dotted decimal notation, for example, 9.99.9.143. (2) Host name.

IPP. Internet Printing Protocol.

IP PrintWay. A component of Infoprint Server for OS/390 that transmits output data sets from the JES spool to printers in a TCP/IP network. Also called PrintWay.

J

JCL. Job control language.

JES. Job entry subsystem.

JES2. An OS/390 subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system. In an installation with more than one processor, each JES2 processor independently controls its job input, scheduling, and output processing.'

JES3. An OS/390 subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system. In complexes that have several loosely coupled processing units, the JES3 program manages processors so that the global processor exercises centralized control over the local processors and distributes jobs to them via a common job queue.

job control language (JCL). A language of control statements used to identify a computer job or describe its requirements to an operating system.

job entry subsystem (JES). An OS/390 subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system.

K

Kanji. A Japanese ideographic alphabet. In Kanji, each character is represented by 2 bytes.

kilobyte (KB). (1) For processor storage, real and virtual storage, and channel volume, 1024 bits. (2) For disk storage capacity and communications volume, 1000 bytes.

L

LAN. local area network.

line data. Data prepared for printing on a line printer such as a 3800 Model 1 Printing Subsystem. Line data is usually characterized by carriage-control characters and table reference characters. Contrast with *MO:DCA-P data*.

line printer daemon (LPD). The printer server that allows other hosts to access its printer.

line printer requester (LPR). A client that allows the local host to submit a data set for printing on a remote printer server.

local area network (LAN). A computer network located on a user's premises within a limited geographical area. Communication within a local area network is not subject to external regulations; however, communication across the LAN boundary may be subject to some form of regulation.

locale. (1) A description of a cultural environment. (POSIX.0). (2) The definition of the subset of a user's environment that depends on language and cultural conventions. (POSIX.2).

logical printer. In NetSpool, the target of the VTAM print data, which acts as the secondary LU (SLU) on the session.

logical unit (LU). A type of VTAM network accessible unit that enables end users to gain access to network resources and communicate with each other.

logon mode. In VTAM, a subset of session parameters specified in a logon-mode table for communication with a logical unit. See also session parameters.

logon-mode table. In VTAM, a set of entries for one or more logon modes. Each logon mode is identified by a logon mode name.

LPD. line printer daemon

LPR. line printer requester

LU. Logical unit.

LU type. The classification of an LU in terms of the specific subset of SNA protocols and options it supports for a given session, namely:

- The mandatory and optional values allowed in the session activation request
- The usage of data stream controls, function management headers, request unit parameters, and sense data values
- Presentation services protocols such as those associated with FMH usage

LU types 0, 1, 2, 3, 4, 6.1, 6.2, and 7 are defined.

M

Management Information Base (MIB). A logical database made up of the configuration, status, and statistical information stored at a device.

megabyte (MB). (1) For processor storage, real and virtual storage, and channel volume, 1 048 576 bytes. (2) For disk storage capacity and communications volume, 1 000 000 bytes.

MIB. See *Management Information Base*.

migration. Activities that relate to the installation of a new version or release of a program to replace an earlier level. Completion of these activities ensures that the applications and resources on your system will function correctly at the new level.

MO:DCA-P data. Print data that has been composed into pages. Text formatting programs can produce composed text data consisting entirely of structured fields.

MVS/ESA™. Multiple Virtual Storage/Enterprise System Architecture.

N

NCP. Network Control Program.

NetSpool. A component of Infoprint Server for OS/390 that allows an installation to automatically direct VTAM application data targeted for a network printer to the JES spool, without changing the VTAM applications. From the JES spool, the data set can be printed on a JES or PSF for OS/390 printer or sent to another location for printing.

network. A collection of data processing products that are connected by communication lines for information exchange between locations.

Network Control Program (NCP). An IBM licensed program that provides communication controller support for single-domain, multiple-domain, and interconnected network capability.

Network Print Facility (NPF). In OS/390 eNetwork™ Communications Server, a feature that routes VTAM, JES2, or JES3 printer output to printers in a TCP/IP network.

NPF. Network Print Facility.

NPM. Network Printer Manager.

Network Printer Manager (NPM) for the Web. IBM Network Printer Manager (NPM) for the Web lets network administrators monitor, control, and configure IBM network printers. NPM also lets network administrators monitor some aspects of printers controlled by PSF for OS/390 and other manufacturers' network printers that comply with RFC 1759.

O

OnDemand. A client/server application that you can use to replace hard copy reports and microfiche, and provide fast, online access to information. An OnDemand server manages electronic archives of reports and documents. You can use an OnDemand client program to search for information and view, print, and FAX copies of reports and documents.

OpenEdition®. See *OS/390 UNIX System Services*.

options data set. In IP PrintWay prior to OS/390 V2R8, a VSAM data set containing one or more options entries. Each options entry contains transmission options used by IP PrintWay to transmit data sets to a print queue in a TCP/IP network. Each options entry can also contain NetSpool parameters that specify data-set characteristics for use by NetSpool.

output writer. A part of the job entry subsystem that receives job output from the system spool.

OS/390 Print Server. A feature of OS/390 V2R5 through V2R7. The second version of the Print Server is called Infoprint Server for OS/390 and runs on OS/390 V2R8 and higher. The OS/390 Print Server supports printing on OS/390 printers, including local printers and remote printers in a TCP/IP network. The OS/390 Print Server allows you to submit print requests from remote workstations in a TCP/IP network, from OS/390 UNIX System Services applications, from batch applications, and from VTAM applications, such as CICS or IMS applications. The OS/390 Print Server consists of the following components:

- OS/390 Print Interface

- NetSpool
- IP PrintWay
- Printing commands for OS/390 UNIX System Services
- IBM AFP Printer Driver for Windows
- IBM AFP Viewer plug-in for Windows
- OS/390 Printer Port Monitor for Windows

OS/390 Printer Port Monitor for Windows. A component of Infoprint Server for OS/390 that runs on a Windows 95 or Windows NT workstation and sends a file for printing to Print Interface running on the OS/390 system.

OS/390 UNIX System Services. OS/390 services that support an environment within which operating systems, servers, distributed systems, and workstations share common interfaces. OS/390 UNIX System Services supports standard application development across multivendor systems. It is required if you want to create and use applications that conform to the POSIX standard. OS/390 UNIX System Services combines the personal power of the workstation, the flexibility of open systems, and the strength of MVS. It supports and fosters a superenvironment of larger operating systems or servers and of distributed systems and workstations that share common interfaces. Users can switch back and forth between the traditional TSO/E interface and the OS/390 UNIX System Services interface. UNIX-skilled users can interact with the system, using a familiar set of standard commands and utilities. MVS-skilled users can interact with the system, using familiar TSO/E commands and interactive menus to create and manage hierarchical file system files and to copy data back and forth between MVS data sets and files. Application programmers and users have both sets of interfaces to choose from and, by making appropriate tradeoffs, can choose to mix these interfaces.

output data set. (1) A data set that a program opens so that it can write to that file. (2) A file that contains the results of processing.

output writer. A part of the job entry subsystem that receives job output from the system spool.

P

page definition. A resource used by PSF that defines the rules of transforming line data into MO:DCA-P data and text controls.

page-format table. In NetSpool prior to OS/390 V2R8, a table that defines page-formatting values NetSpool uses for SCS data streams that do not contain SHF (Set Horizontal Format) or SVF (Set Vertical Format) commands. The table can contain several entries, each entry containing a different set of page-formatting values.

parameter. Information that the user supplies to a panel, command, or function.

partitioned data set (PDS). A data set in direct access storage that is divided into partitions, called members, each of which can contain a program, part of a program, or data. Synonymous with program library.

pathname. (1) A filename specifying all directories leading to the file. (2) A filename specifying all directories leading to a file plus the filename itself. (3) A string that is used to identify a file. A pathname consists of, at most, (PATH_MAX) bytes, including the terminating null character. It has an optional beginning slash, followed by zero or more filenames separated by slashes. If the pathname refers to a directory, it may also have one or more trailing slashes. Multiple successive slashes are considered to be the same as one slash. A pathname that begins with two successive slashes may be interpreted in an implementation-defined manner, although more than two leading slashes shall be treated as a single slash. (POSIX.1). In the OS/390 UNIX System Services implementation, the C/370™ functions **fopen()**, **freopen()**, **remove()**, and **rename()** interpret names with exactly two leading slashes, no leading blanks or other characters, and the third character not a slash to mean that the rest of the name refers to a traditional MVS data set.

pel. Picture element.

Picture element (pel, pixel). (1) In computer graphics, the smallest element of a display surface that can be independently assigned color and intensity. (T). (2) The area of the finest detail that can be reproduced effectively on the recording medium. (3) An element of a raster pattern about which a toned area on a photoconductor can appear. (4) The addressable unit on a 3800 Printing System Model 3 or 8.

PIDU. Printer Inventory Definition Utility.

PLU. Primary logical unit.

port. (1) A part of the system unit or remote controller to which cables for external devices (display stations, terminals, or printers) are attached. The port is an access point for data entry or exit. (2) A specific communications end point within a host. A port is identified by a port number.

POSIX. Portable Operating System Interface for Computer Environments, an interface standard governed by the IEEE and based on UNIX. POSIX is not a product; rather, it is an evolving family of standards describing a wide spectrum of operating system components ranging from C language and shell interfaces to system administration.

PostScript. A page description language with graphics capabilities that was developed by Adobe Systems, Incorporated.

primary logical unit (PLU). In SNA, the logical unit (LU) that sends the BIND to activate a session with its partner LU. Contrast with *secondary logical unit*.

print queue. A list of items waiting to be printed.

Print Services Facility™ (PSF). A licensed program that manages and controls the input data stream and output data stream required by supported IBM page printers. PSF combines print data with other resources and printing controls to produce AFP output.

Print Interface. A component of the OS/390 Print Server that accepts input from remote workstations that have TCP/IP access and from OS/390 UNIX System Services printing commands and creates output data sets on the JES spool.

Print Server. See *OS/390 Print Server*.

printer definition. In Infoprint Server for OS/390, an entry in the Printer Inventory that contains information about an OS/390 printer or set of printers that share the same characteristics. A printer definition contains information that Infoprint Server uses to print files.

Printer Inventory. In Infoprint Server for OS/390, a set of files that contain printer definitions for OS/390 printers. Each printer definition is identified with a unique printer name; the job submitter selects the printer name when printing a file. The printer definitions contain information that Infoprint Server for OS/390 uses to print files.

Printer Inventory Definition Utility. In Infoprint Server for OS/390, a utility program that creates objects in the Printer Inventory.

Printer Port Monitor. See *OS/390 Printer Port Monitor for Windows*.

PrintWay. See *IP PrintWay*.

protocol. A set of semantic and syntactic rules that determines the behavior of functional units in achieving communication.

PSF. Print Services Facility.

PSF/6000. An intelligent printer driver that provides AFP capabilities for the AIX operating system on the RS/6000 system. AFP capabilities include electronic forms, images, graphics, and typographical fonts. Also called PSF for AIX.

R

RACF. Resource Access Control Facility

Request for Comments (RFC). In Internet communications, the document series that describes a

part of the Internet suite of protocols and related experiments. All Internet standards are documented as RFCs.

request unit (RU). A message unit that contains control information, end-user data, or both.

resource. A collection of printing instructions used by Print Services Facility in addition to the print data set, to produce the printed output. PSF resources include coded fonts, font character sets, code pages, page segments, overlays, form definitions, and page definitions.

Resource Access Control Facility (RACF). An IBM-licensed product that provides for access control by identifying and verifying users to the system, authorizing access to protected resources, logging detected unauthorized attempts to enter the system, and logging detected accesses to protected resources.

response unit (RU). A message unit that acknowledges a request unit. It may contain prefix information received in a request unit. If positive, the response unit can contain additional information (such as session parameters in response to BIND SESSION). If negative, the response unit contains sense data defining the exception condition.

Restructured Extended Executor (REXX). A general-purpose, procedural language for end-user personal programming, designed for ease by both casual general users and computer professionals. It is also useful for application macros. REXX includes the capability of issuing commands to the underlying operating system from these macros and procedures. Features include powerful character-string manipulation, automatic data typing, manipulation of objects familiar to people, such as words, numbers, and names, and built-in interactive debugging.

retain time. In IP PrintWay, the length of time to keep a data set on the JES spool after either a successful transmission to the destination or a failed transmission, after retrying the transmission the number of times specified in the retry limit. You can specify a retain time for 2 different situations:

- Retain time for data sets that have been successfully transmitted
- Retain time for data sets whose transmission has failed

retry limit. In IP PrintWay, the maximum number of retries that IP PrintWay is to attempt.

retry time. In IP PrintWay, the time between two attempts to send the data set to its destination.

REXX. Restructured Extended Executor

RFC. Request for Comments.

routing data set. In IP PrintWay prior to OS/390 V2R8, a VSAM data set containing a routing entry for each print queue to which IP PrintWay can transmit output data sets. Each entry contains the name of the remote print queue, the IP address or name of the print queue's host system, the name of an options entry, and other routing information. Each routing entry can also define a NetSpool logical printer.

RU. Request/response unit.

S

SBCS. Single-byte character set.

SCS. SNA Character String.

SDSF. System Display and Search Facility.

secondary logical unit (SLU). In SNA, the logical unit (LU) that receives the BIND request to establish a session with its partner LU. Contrast with *primary logical unit*.

sense code. In SNA, the data sent with a negative response, indicating the reason for the response.

sequential data set. (1) A data set whose records are organized on the basis of their successive physical positions, such as on magnetic tape. (2) A data set in which the contents are arranged in successive physical order and are stored as an entity. The data set can contain data, text, a program, or part of a program. Contrast with *partitioned data set (PDS)*.

server. (1) On a network, the computer that contains the data or provides the facilities to be accessed by other computers on the network. (2) A program that handles protocol, queuing, routing, and other tasks necessary for data transfer between devices in a computer system.

Server Message Block (SMB). (1) A protocol for remote file and print access used by Windows clients. This protocol is also known as Common Internet File System (CIFS). (2) A program that handles protocol, queuing, routing, and other tasks necessary for data transfer between devices in a computer system.

session. A logical connection between two network accessible units that can be activated, tailored to provide various protocols, and deactivated, as requested.

session parameters. In SNA, the parameters that specify or constrain the protocols, for a session between two network addressable units (NAUs).

shell script. A file of shell commands. If the file is executable; a user can run it by specifying the file's name as a shell command or as an operand on **sh** or on the TSO/E OMVS command. A shell script is like a TSO/E REXX program.

shift-out, shift-in (SOSI). Special EBCDIC or ASCII characters in the data stream that indicate switches between double-byte and single-byte fonts.

Simple Network Management Protocol (SNMP). A protocol that enables a management station to configure, monitor, and receive trap messages from network devices.

single-byte character set (SBCS). A set of characters in which each character is represented by a one-byte code. Contrast with *double-byte character set*.

SLU. Secondary logical unit.

SMB. Server Message Block.

SMF. System Management Facilities.

SNA. Systems Network Architecture.

SNA Character String (SCS). In SNA, a character string composed of EBCDIC controls, optionally intermixed with end-user data, that is carried within a request/response unit.

SNMP. See *Simple Network Management Protocol*.

SNMP agent. Software that enables a device to respond to manager requests to view or update Management Information Base (MIB) data, and send traps reporting problems or significant events.

SNMP Manager. In SNMP, software in a network management station that enables the station to send requests to view or update MIB variables, to send and receive inform-requests, and to receive traps from an agent.

SOSI. See *shift-out, shift-in*.

spool. Simultaneous peripheral operation online.

startup procedure. JCL for a procedure to start an application (or, NetSpool and IP PrintWay).

superuser. A system user who operates without restrictions. A superuser has the special rights and privileges needed to perform administrative tasks.

syntax. The grammatical rules for constructing a command.

System Display and Search Facility (SDSF). An IBM-licensed program that provides a menu-driven full screen interface to obtain detailed information about the jobs and resources in an MVS/JES2 system.

System Management Facilities (SMF). An optional control program feature of OS/390 that provides the means for gathering and recording information that can be used to evaluate system usage.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through, and controlling the configuration and operation of, networks.

T

table reference character (TRC). A numeric character corresponding to the order in which font character sets have been specified with the **chars** job attribute or in the page definition used to print a job. It is used to select a font character set during printing.

TCP. Transmission Control Protocol.

TCP/IP. Transmission Control Protocol/Internet Protocol.

Telnet. In the Internet suite of protocols, a protocol that provides remote terminal connection service. It allows users of one host to log on to a remote host and interact as directly attached terminal users of that host.

Time Sharing Option (TSO). An operating system option that provides interactive time sharing from remote terminals.

trace. A record of the execution of a computer program. It exhibits the sequences in which the instructions were executed. (A)

transform. A program that converts a data stream from one format to another, for example, from PCL to AFP, PDF to AFP, and so on. The IBM-provided transforms are implemented as DLL filters.

Transmission Control Protocol (TCP). A communications protocol used in Internet and in any network that follows the U.S. Department of Defense standards for inter-network protocol. TCP provides a reliable host-to-host protocol between hosts in packet-switched communications networks and in interconnected systems of such networks. It assumes that the Internet protocol is the underlying protocol.

transmission-queue data set. In IP PrintWay, a data set containing an entry for each data set that IP PrintWay is to transmit to the remote system or that IP PrintWay is retaining on the JES spool.

transparent data. (1) Data that is of no significance to the receiver. (2) Data that can contain any hexadecimal value.

trap. A message that reports a problem or a significant event.

TRC. Table reference character.

U

UCS. Universal character set.

universal character set (UCS). A printer feature that permits the use of a variety of character arrays. Synonymous with *font*.

UNIX. A highly portable operating system originally developed by Bell Laboratories that features multiprogramming in a multiuser environment. UNIX is implemented in the C language. UNIX was originally developed for use on minicomputers but has been adapted on mainframes and microcomputers. It is especially suitable for multiprocessor, graphics, and vector-processing systems. Many of the commands in the OS/390 UNIX System Services shell are based on similar commands available with UNIX System V.

UNIX filter. A filter that accepts input via **stdin**, and returns the output via **stdout**. Options and positional arguments can be specified. UNIX filters may be any shell executable, for example, a shell script, a REXX exec, a C program, and so on. See also filter. Contrast with DLL filter.

UNIX System Services. See *OS/390 UNIX System Services*.

user port. In IP PrintWay, a port address that is outside the range of addresses defined in RFC 1179 for the LPR source port.

V

Virtual Telecommunications Access Method (VTAM). An IBM licensed program that controls communication and the flow of data in an SNA network. It provides single-domain, multiple-domain, and interconnected network capability.

VTAM. Virtual Telecommunications Access Method.

W

Workbench for OS/2 and Windows. An application that runs under Windows or WIN-OS/2® that enables you to browse and print AFP documents and resources on your workstation.

Bibliography

This section lists publications that may be helpful to you as you configure and use Infoprint Server for OS/390.

Infoprint Server for OS/390

Title	Order Number
<i>OS/390 Infoprint Server Customization</i>	G544-5694
<i>OS/390 Infoprint Server Introduction</i>	G544-5696
<i>OS/390 Infoprint Server Messages and Diagnosis</i>	G544-5690
<i>OS/390 Infoprint Server Migration</i>	G544-5697
<i>OS/390 Infoprint Server Operation and Administration</i>	S544-5693
<i>OS/390 Infoprint Server User's Guide</i>	S544-5692

Print Services Facility for OS/390

Title	Order Number
<i>AFP Conversion and Indexing Facility: User's Guide</i>	S544-5285
<i>PSF for OS/390: Customization</i>	S544-5622
<i>PSF for OS/390: Diagnosis</i>	G544-5623
<i>PSF for OS/390: Download for OS/390</i>	S544-5624
<i>PSF for OS/390: Introduction</i>	G544-5625
<i>PSF for OS/390: Messages and Codes</i>	G544-5627
<i>PSF for OS/390: User's Guide</i>	S544-5630

Advanced Function Presentation (AFP)

Title	Order Number
<i>AFP: Printer Information</i>	G544-3290
<i>AFP: Printer Summary</i>	G544-3135
<i>AFP: Programming Guide and Line Data Reference</i>	S544-3884
<i>IBM AFP Fonts: Font Summary</i>	G544-3810
<i>IBM AFP Fonts: Font Summary for AFP Font Collection</i>	S544-5633
<i>IBM Data Stream and Object Architectures: Bar Code Object Content Architecture Reference</i>	S544-3766

OS/390 Version 2 Release 8

Title	Order Number
<i>OS/390 C/C++ Programming Guide</i>	SC09-2362
<i>OS/390 Distributed File Service SMB Administration Guide and Reference</i>	SC24-5882

Title	Order Number
<i>OS/390 ISPF Dialog Developer's Guide and Reference</i>	SC28-1273
<i>OS/390 JES2 Commands</i>	GC28-1790
<i>OS/390 JES2 Initialization and Tuning Guide</i>	SC28-1791
<i>OS/390 JES2 Initialization and Tuning Reference</i>	SC28-1792
<i>OS/390 JES3 Commands</i>	<i>OS/390 JES3 Commands</i>
<i>OS/390 JES3 Initialization and Tuning Guide</i>	SC28-1802
<i>OS/390 JES3 Initialization and Tuning Reference</i>	SC28-1803
<i>OS/390 MVS JCL Reference</i>	GC28-1757
<i>OS/390 MVS Product Management</i>	GC28-1730
<i>OS/390 MVS Programming: Authorized Assembler Services Reference ALE-DYN</i>	GC28-1764
<i>OS/390 MVS Programming: Authorized Assembler Services Reference ENF-IXG</i>	GC28-1765
<i>OS/390 MVS Programming: Authorized Assembler Services Reference LLA-SDU</i>	GC28-1766
<i>OS/390 MVS Programming: Authorized Assembler Services Reference SET-WTO</i>	GC28-1767
<i>OS/390 Planning for Installation</i>	GC28-1726
<i>OS/390 SDSF Guide and Reference</i>	SC28-1622
<i>OS/390 Security Server (RACF) General User's Guide</i>	SC28-1917
<i>OS/390 Security Server (RACF) Security Administrator's Guide</i>	SC28-1915
<i>OS/390 Summary of Message Changes</i>	GC28-1499
<i>OS/390 UNIX System Services Command Reference</i>	SC28-1892
<i>OS/390 UNIX System Services User's Guide</i>	SC28-1891

OS/390 SecureWay Communications Server

Title	Order Number
<i>OS/390 eNetwork Communications Server: SNA Programming</i>	SC31-8573
<i>OS/390 SecureWay Communications Server: IP Application Programming Interface Guide</i>	SC31-8516
<i>OS/390 SecureWay Communications Server: IP Configuration</i>	SC31-8513
<i>OS/390 SecureWay Communications Server: IP Migration</i>	SC31-8512
<i>OS/390 SecureWay Communications Server: SNA Diagnosis V1 Techniques and Procedures</i>	LY43-0079
<i>OS/390 SecureWay Communications Server: SNA Diagnosis V2 FFST Dumps and the VIT</i>	LY43-0080
<i>OS/390 SecureWay Communications Server: SNA Messages</i>	SC31-8569
<i>OS/390 SecureWay Communications Server: SNA Network Implementation</i>	SC31-8563
<i>OS/390 SecureWay Communications Server: SNA Operation</i>	SC31-8567
<i>OS/390 SecureWay Communications Server: SNA Resource Definition Reference</i>	SC31-8565
<i>Systems Network Architecture: Sessions Between Logical Units</i>	GC20-1868

CICS for OS/390

Title	Order Number
<i>CICS Customization Guide</i>	SC33-1683
<i>CICS Diagnosis Reference</i>	LY33-6088
<i>CICS Resource Definition Guide</i>	SC33-1684
<i>CICS Supplied Transactions</i>	SC33-1686

IMS/ESA[®] Release 5

Title	Order Number
<i>IMS/ESA Application Programming: EXEC DLI Commands for CICS and IMS</i>	SC26-8726
<i>IMS/ESA Administration Guide: System</i>	SC26-8730

3270 Data Stream

Title	Order Number
<i>IBM 3270 Kanji Data Streams</i>	GA18-2980

Index

Special Characters

% operator in filter options 30
/etc/Printsrv/aopd.conf, creating 19
/etc/Printsrv directory, creating 18
/etc/profile, setting environment variables in 20
/usr/lpp/Printsrv/bin, specifying in PATH variable 21, 142
/var/Printsrv directory, creating 18

Numerics

00000000.tab file 59
01200000.tab file 59
11000000.tab file 59
161 and 162 ports, unrestricting in TCP/IP data set 29
3270 data stream support 10
40010000.tab file 59
40020000.tab file 59
515 port number
 specified as default 36
 unrestricting in TCP/IP data set 29
631 port number
 specified as default 38
 unrestricting in TCP/IP data set 29
80000000.tab file 59

A

ABAP data, converting 52
abend D37 65, 87
address spaces, running NetSpool in multiple 66
administrator group, for security 22
AFP Printer Driver 128
 downloading 128
 installing 128
AFP Printer Driver for Windows 127
AFP Printer Driver for Windows, overview 4
AFP Viewer plug-in 128
 downloading 128
 installing 128
AFP Viewer plug-in for Windows 127
AFP Viewer plug-in for Windows, overview 4
afpdv95.exe 127
afpviewr.exe 127
AIX, transforms on 44
ANFCALL DD statement 90
ANFLIB DD statement 90
ANFMFILE program 65, 87
ANFMJCL sample JCL 87
ANFMMSG DD statement 90
ANFQUEUE DD statement 90
ANFUBTBL, tables for block letters 123
ANFUEXTTP control block 97, 103, 107, 111, 115, 118
ANFULINK sample JCL 124
ANFUXRSP exit 119

ANFUXxxx exits, sample 124
ANFWPROC startup procedure 88
AOP_HORIZONTAL_MARGINS
 environment variable 47, 137
AOP_PAGE_HEIGHT environment variable 47, 137
AOP_PAGE_WIDTH environment variable 47, 137
AOP_RESOLUTION environment variable 47, 137
AOP_SAP2AFP_RESOURCES
 environment variable 49, 141
AOP_VERTICAL_MARGINS
 environment variable 47, 137
AOPADMIN group 22
AOPCONF environment variable 20, 141
aopd.conf configuration file
 ascii-codepage keyword 20, 131
 base-directory keyword 20, 131
 creating 19
 ebcdic-codepage keyword 20, 131
 format 131
 inventory keyword 19, 132
 ipp-port-number keyword 38
 job-prefix keyword 29, 132
 lpd-port-number keyword 36, 132
 naming in AOPCONF environment variable 20
 snmp-community keyword 62, 132
 start-daemons keyword 19, 36, 39, 49, 62, 133
aopd.conf See also configuration file, aopd.conf 131
aopfiltr.so 34
 source code 31
AOPINIT EXEC 25
AOPOPER group 22
AOPOPTIONS environment variable 34
AOPPATH environment variable 20, 141
AOPPRINT JCL procedure, customizing 35
aopsetup shell script 22
AOPSTART procedure 22
AOPSTOP procedure 23
aopxfg.conf configuration file 44
AOPXFD_CONF environment variable 49, 141
aopxfg.conf See also configuration file, aopxfg.conf 135
APF authorization, for IP PrintWay IPP client 78
APF authorization, for Print Interface IPP server 39
APF-authorizing NetSpool exits 70
APF-authorizing SYS1.IMAGELIB 86
APIPJCL startup procedure for NetSpool 67
APIJPTD1 75
APIJPTD2 75
APIMIJCL sample JCL 65

APIMMSG DD statement 69
APIPP1X2 control block 70
APIPP1X2 macro 75
APIPPTD1 exit, NetSpool 72
APIPPTD2 exit, NetSpool 73
ascii-codepage keyword in configuration file 20, 131
attribute types, 3270 data streams 146
attributes, job, specifying in environment variable 34
AUTOSTOP parameter (JES2) 81

B

backing up the Printer Inventory 25
banner page, adding in exit 102, 111
banner page, controlling in sample exit 124
bar codes, mapping to BCOCA for SAP to AFP transform 50
barcode.tab configuration file 50
base-directory keyword in configuration file 20, 131
Begin Data Set exit, IP PrintWay examples 106
 input 102
 installing 124
 modifying 125
 output 105
 overview 102
 programming considerations 92
 sample 124
Beginning of File exit, NetSpool (APIPPTD1) 72
 examples 73
 input 72
 installing 76
 modifying 76
 output 73
 programming considerations 69
 sample 75
binary data support 10
blank truncation 80, 83
BLNKTRNC parameter (OUTCLASS) 80
Block Letter routine
 description of 122
 input 123
 link-editing with exit 124
 output 124
 sample exit for 124
 tables (ANFUBTBL) 123
BPXPRMxx member, updating 18

C

cfilter.c 31
CICS, customization required to print from 15
class
 changing in exit 96
 CLASS parameter (JES2) 81

class (*continued*)

- QUEUE parameter (JES2) 81
- specifying as JES2 work-selection criteria 81, 82
- specifying as JES3 work-selection criteria 85
- WC parameter (JES3) 85

classes, for PCL to AFP transform 44

CLASSPATH environment variable 39, 141

code pages, default, specifying in configuration file 20

code points

- 3270 data streams 145
- SCS 143

coexistence, regarding Printer Inventory 17

command codes, 3270 data streams 145

community name for SNMP in configuration file 62, 132

configuration file, aopd.conf

- ascii-codepage keyword 20, 131
- base-directory keyword 20, 131
- creating 19
- ebcdic-codepage keyword 20, 131
- format 131
- inventory keyword 19, 132
- ipp-port-number keyword 38
- job-prefix keyword 29, 132
- lpd-port-number keyword 36, 132
- naming in AOPCONF environment variable 20
- snmp-community keyword 62, 132
- start-daemons keyword 19, 36, 39, 49, 62, 133

configuration file, aopxfd.conf 44

- creating 44
- environment keyword 47, 137
- example 47
- format 45, 135
- maximum-active keyword 47, 137
- maximum-idle-time keyword 46, 137
- minimum-active keyword 47, 137
- start-command keyword 45, 136
- start-directory keyword 46, 136
- transform keyword 45, 135

configuration file, barcode.tab 50

configuration file, defcp.tab 52

configuration file, fonts.tab 54

configuration file, image.tab 56

configuration file, pagedef.tab 57

configuration file, xxxx0000.tab 59

configuration files for SAP to AFP transform 50

CONSNAM parameter, for NetSpool 68

consoles, for NetSpool 66, 68

control codes, 3270 data streams 145

COPIES parameter, changing in exit 96

customer fonts 55

customizing

- AFP Printer Driver for Windows 127
- AFP Viewer plug-in for Windows 127
- AOPPRINT JCL procedure 35
- Infoprint Server Transforms for OS/390 43

customizing (*continued*)

- IP PrintWay 77
- IP PrintWay IPP client 78
- LPD 35
- NetSpool 65
- OS/390 SNMP agent 61
- Print Interface 27
- Print Interface IPP server 38
- Printer Inventory Manager 17
- Printer Port Monitor for Windows 127
- SNMP manager 63
- SNMP subagent 61
- Transform Manager 43
- Windows client 127

D

D37 abend 65, 87

daemons to start 19, 36, 39, 49, 62

DATASETPREFIX parameter, in TCPIP.DATA data set 78

default printer, specifying 34, 142

defcp.tab configuration file 52

destination

- changing in exit 96
- DGROUP parameter (JES3) 84
- ROUTECD parameter (JES2) 82
- specifying as JES2 work-selection criteria 82
- specifying as JES3 work-selection criteria 85

DEVICE statement (JES3) 84

DGROUP parameter (JES3) 84

diagram of system flow 5

directories, creating 18

directory name for files, specifying in configuration file 20, 131

DLLs, specifying directory for 21, 142

downloading AFP printer driver 128

downloading AFP Viewer plug-in 128

downloading Windows client 127

DTYPE parameter (JES3) 84

dynamic enablement of Infoprint Server 17

dynamic enablement of Infoprint Server Transforms 44

DYNAMIC parameter (JES3) 84

E

ebcdic-codepage keyword in configuration file 20, 131

enabling Infoprint Server 17

enabling Infoprint Server Transforms 44

End Data Set exit, IP PrintWay

- examples 114
- input 111
- installing 124
- modifying 125
- output 113
- overview 111
- programming considerations 92
- sample 124

environment keyword in transform configuration file 47, 137

environment variables

- AOP_HORIZONTAL_MARGINS variable 47, 137
- AOP_PAGE_HEIGHT variable 47, 137
- AOP_PAGE_WIDTH variable 47, 137
- AOP_RESOLUTION variable 47, 137
- AOP_SAP2AFP_RESOURCES variable 49, 141
- AOP_VERTICAL_MARGINS variable 47, 137
- AOPCONF variable 20, 141
- AOPOPTIONS variable 34, 141
- AOPPATH variable 20, 141
- AOPXFD_CONF variable 49, 141
- CLASSPATH variable 39, 141
- for IPP server 39
- for PCL to AFP transforms 47, 137
- for Print Interface IPP server 39
- for Printer Inventory Manager 20
- for transforms 49
- JAVA_HOME variable 39, 141
- LIBPATH variable 21, 142
- LPDEST variable 34, 142
- MANPATH variable 21, 142
- NLSPATH variable 21, 142
- PATH variable 21, 142
- PRINTER variable 34, 142
- summary of 141

examples

- Begin Data Set exit 106
- End Data Set exit 114
- JES2 initialization 80
- JES3 initialization 83
- Message exit 118
- Record exit 110
- Routing exit 101
- SMF exit 117
- transform configuration file 47

exit programs 69, 90

- APF-authorizing NetSpool exits 69
- APIPP1X2 control block 70
- installing IP PrintWay exits 124
- installing NetSpool exits 76
- IP PrintWay Begin Data Set exit 102
- IP PrintWay End Data Set exit 111
- IP PrintWay Message exit 117
- IP PrintWay Record exit 106
- IP PrintWay Response Notification exit 119
- IP PrintWay Routing exit 96
- IP PrintWay SMF exit 115
- modifying IP PrintWay exits 125
- modifying NetSpool exits 76
- NetSpool Beginning of File exit 72
- NetSpool Transparent Data Control exit 73
- order of invocation, IP PrintWay 91
- Print Interface filters 30
- programming considerations, IP PrintWay 92
- programming considerations, NetSpool 69
- sample IP PrintWay exits 124
- sample NetSpool exits 75
- STEPLIB for NetSpool exits 69

exporting the Printer Inventory 25

F

- FACILITY class 22
- FCB, APF authorization for
SYS1.IMAGELIB 86
- FCB work-selection criterion 85
- filter programs, Print Interface 30
 - examples of % operator 31
 - input 32
 - modifying 33
 - options for 30
 - output 32, 33
 - programming considerations 33
 - REXX examples 31
 - sample 34
 - source code 31
 - testing 33
 - transform filters (IBM) 34
 - writing a DLL filter 31
 - writing a UNIX filter 32
- fonts.tab file 54
- form definitions, mapping to PJPAPER
values 57
- form name
 - changing in exit 96
 - FORMS parameter (JES2) 82
 - specifying as JES2 work-selection
criteria 82
 - specifying as JES3 work-selection
criteria 85
- FSS initialization statement (JES2) 81
- FSS initialization statement (JES3) 83
- FSSNAME parameter (JES3) 83, 84
- functional subsystem (FSS), IP PrintWay
 - defining multiple FSAs 79
 - defining multiple FSSs 79
 - JES2 initialization statements 80
 - JES3 initialization 83
 - name, specifying in JES2 81
 - name, specifying in JES3 83
 - planning 79
- functional subsystem application (FSA),
IP PrintWay
 - determining work-selection
criteria 80
 - how many IP PrintWay FSAs to
create 79
 - name, specifying in JES2 81
 - name, specifying in JES3 84

H

- HFS directories, creating 18
- HRI character 50
- human readable interface (HRI)
character 50

I

- IAZJSPA control block 92
- IBM-1047, specified as default 20, 131
- IDCAMS parameters to allocate
transmission-queue data sets 87
- image data in SAP to AFP transform 56
- image.tab file 56
- IMS, customization required to print
from 15

- Infoprint Server Transforms for OS/390
 - customizing 43
 - transform DLLs 34
- Infoprint Server Transforms for OS/390,
overview 7
- input data streams, types supported 9
- installing
 - NetSpool exits 76
- installing AFP printer driver 128
- installing AFP Viewer plug-in 128
- installing IP PrintWay exits 124
- interfaces, programming 30
- internet, downloading Windows client
from 127
- INTERVAL parameter 28
- INV parameter in IP PrintWay startup
procedure 90
- INV parameter in NetSpool startup
procedure 68
- Inventory, Printer
 - backing up 25
 - naming in configuration file 19
 - naming in IP PrintWay startup
procedure 90
 - naming in NetSpool startup
procedure 68
- inventory keyword in configuration
file 19, 132
- IP address, changing in exit 96
- IP PrintWay
 - APF-authorizing SYS1.IMAGELIB 86
 - customizing 77
 - exits, writing 90
 - FSS, defining 79
 - IPP client, customizing 78
 - sharing AFP printers with PSF for
OS/390 125
 - startup procedure for 88
 - TCPIP.DATA data set, customizing
for 77
 - transmission-queue data set,
allocating 86
- IP PrintWay, overview 10
- IPP client in IP PrintWay,
customizing 78
- IPP client in IP PrintWay, DD
statements 90
- ipp-port-number keyword in
configuration file 38
- IPP server in Print Interface,
customizing 38
 - APF authorizing libraries 39
 - environment variables for 39
 - IPP workstation client,
customizing 39
 - specifying port number for 38
 - starting daemon for 39
 - URI used by IPP workstation
client 39
- ISO8859-1, specified as default 20, 131
- ISPF panels, Infoprint Server
 - defining libraries in TSO logon
procedure 24
 - displaying Infoprint Server option 24
 - enabling 24
 - modifying AOPINIT EXEC 25

J

- JAVA_HOME environment variable 39,
141
- JES work-selection criteria
 - determining 80
 - specifying in JES2 82
 - specifying in JES3 85
- JES2
 - FSS statement example 81
 - initialization statements 80
 - PRTnnnnn statement 81
- JES3
 - DEVICE statement 84
 - FSS statement 83
 - initialization statements 83
- JESNEWS data set
 - indicator in ANFUEXTP control
block 94
 - suppressing SMF type-6 record
for 115, 124
- JNAME parameter (JES3) 84
- job attributes, specifying in environment
variable 34
- job control language (JCL)
 - sample to link-edit exit 124
 - startup, for Infoprint Server
daemons 22
 - startup, IP PrintWay 89
 - startup, NetSpool 66
- job-prefix keyword in configuration
file 29, 132
- job separator page data area
(IAZJSPA) 92
- JSPA control block 92

L

- LANGUAGE parameter, in NetSpool
startup procedure 68
- LIBPATH environment variable 21, 142
- libraries, defining in TSO logon
procedure (Print Interface) 24
- logon procedure for ISPF panels 24
- LPD, customizing 35
 - command codes supported in 36
 - specifying port number for 36
 - starting daemon 36
- lpd-port-number keyword in
configuration file 36, 132
- LPDEST environment variable 34, 142
- LUCLASS parameter in NetSpool startup
procedure 67

M

- man pages, specifying directory for 21,
142
- MANPATH environment variable 21,
142
- margins, specifying for PCL to AFP
transforms 47, 137
- maximum-active keyword in transform
configuration file 47, 137
- maximum-idle-time keyword in
transform configuration file 46, 137
- MCS console 66

- Message exit, IP PrintWay
 - examples 118
 - input 118
 - installing 124
 - modifying 125
 - output 118
 - overview 117
 - programming considerations 92
 - sample 124
- message-log data set, IP PrintWay
 - allocating space 87
 - defining in startup procedure 90
 - initializing 87
- message-log data set, NetSpool 65
- migration program, overview 4
- minimum-active keyword in transform configuration file 47, 137
- MODE parameter
 - in DEVICE statement (JES3) 84
 - in PRTnnnnn statement (JES2) 82
- MSGDEST parameter (JES3) 83

N

- NetSpool
 - customizing 65
 - exits for transparent data, writing 69
 - LU classes, specified in startup procedure 67
 - message log, allocating 65
 - startup procedure for 66
- NetSpool, overview 8
- Network Printer Manager (NPM) for the Web, customizing 63
- network printers, customizing PSF for OS/390 for 125
- NLSPATH environment variable 21, 142
- NLVs for NetSpool 66
- NPM for the Web, customizing 63

O

- operator group, for security 22
- OS/390 SNMP agent, customizing and starting 61
- OTF data, converting fonts in 54
- OUTCLASS statement (JES2) 80
- overview of Infoprint Server components 1
 - AFP Printer Driver for Windows 4
 - AFP Viewer plug-in for Windows 4
 - Infoprint Server Transforms for OS/390 7
 - IP PrintWay 10
 - NetSpool 8
 - Print Interface 4
 - Printer Inventory 3
 - Printer Inventory Manager 3
 - Printer Port Monitor for Windows 4
 - Transform Manager 7
 - Windows client 4

P

- page definitions, mapping to PJPAPER values 57

- page height, specifying for PCL to AFP transforms 47, 137
- page width, specifying for PCL to AFP transforms 47, 137
- pagedef.tab file 57
- panels, ISPF
 - defining libraries in TSO logon procedure 24
 - displaying Infoprint Server option 24
 - enabling 24
 - modifying AOPINIT EXEC 25
- parameters
 - JES2 initialization statements 80
 - JES3 initialization statements 83
 - LANGUAGE parameter 68
 - TRACE parameter 68
- PATH environment variable 21, 142
- PCL to AFP transform, customizing 43
- PCL to AFP transform classes 44
- PCL to AFP transforms, environment variables for 47, 137
- PDEFAULT parameter (JES3) 85
- PDF to AFP transform, customizing 43
- performance and the TRKCELL parameter (JES2) 82
- PIDU, used to backup Printer Inventory 25
- PJPAPER, mapping to page and form definitions 57
- PNAME parameter (JES3) 83
- Port Monitor for Windows, overview 4
- port number
 - for Print Interface IPP server 38
 - for Print Interface LPD 36
 - used by LPR 36
- ports, unrestricting in TCP/IP 28, 29
- PostScript to AFP transform, customizing 43
- prefix for job identifier 29, 132
- PRESELECT parameter (JES2) 82
- Print Interface
 - AOPPRINT JCL procedure, customizing 35
 - configuration file statements 29
 - customizing 27
 - environment variables for UNIX commands 34
 - filters, writing 30
 - general customization required for all functions 27
 - IPP server customization 38
 - JES customization 29
 - LPD customization 35
 - TCP/IP customization 28
- Print Interface, overview 4
- print queue, changing in exit 96
- printer, setting a default 34, 142
- PRINTER environment variable 34, 142
- Printer Inventory
 - backing up 25
 - naming in configuration file 19
 - naming in IP PrintWay startup procedure 90
 - naming in NetSpool startup procedure 68
- Printer Inventory, overview 3

- Printer Inventory Manager
 - backing up Printer Inventory 25
 - creating configuration file for 19
 - creating HFS directories for 18
 - creating startup and shutdown procedures for 22
 - customizing 17
 - dynamic enablement of Infoprint Server 17
 - enabling ISPF panels for 24
 - establishing security for 21
 - making run-time libraries available for 18
 - setting environment variables for 20
- Printer Inventory Manager, overview 3
- Printer Port Monitor, comparing to SMB 127
- Printer Port Monitor for Windows, downloading 127
- Printer Port Monitor for Windows, overview 4
- printer setup string
 - adding in a filter program 30
- printer setup string, adding in exit 102
- printer sharing, customizing PSF for OS/390 for 125
- PrintWay
 - APF-authorizing SYS1.IMAGELIB 86
 - customizing 77
 - exits, writing 90
 - FSS, defining 79
 - IPP client, customizing 78
 - sharing AFP printers with PSF for OS/390 125
 - startup procedure for 88
 - TCPIP.DATA data set, customizing for 77
 - transmission-queue data set, allocating 86
- PrintWay, overview 10
- PROC parameter (JES2) 81
- procedure, startup
 - for Infoprint Server daemons 22
 - IP PrintWay 88
 - sample 89
 - specifying name of in JES2 81
 - specifying name of in JES3 83
- NetSpool 66
- sample 67
- profile, TCP/IP 28
- programming considerations for filter programs 33
- programming considerations for IP PrintWay exits 92
- programming considerations for NetSpool exit programs 69
- programming interfaces 30
- PRTnnnnn initialization statement (JES2) 81
- PRTOPTNS parameter, changing in exit 96
- PRTQUEUE parameter, changing in exit 96
- PS, default job prefix 29, 132
- PSF for OS/390, customization required to use Printer Inventory 15

PSF for OS/390, customizing for printer sharing 125
 PSF for OS/390, enabling SNMP reporting in Printer Inventory 62
 PSF for OS/390, roadmap of components to customize 15
 publications, related 161

Q

QUEUE parameter (JES2) 81

R

RACF 21, 87
 Record exit, IP PrintWay
 examples 110
 input 107
 installing 124
 modifying 125
 output 109
 overview 106
 sample 124
 record format
 input to Begin Data Set exit 104
 input to End Data Set exit 113
 input to Routing exit 98
 region size, IP PrintWay 89
 region size, NetSpool 67
 registers
 for Begin Data Set exit 102
 for Beginning of File exit 72
 for Block-Letter program 123
 for End Data Set exit 111
 for Message exit 118
 for Record exit 107
 for SMF exit 115
 for the Routing exit 96
 for Transparent Data exit 74
 remote transforms 44
 resolution, specifying for PCL to AFP transforms 47, 137
 Resource Access Control Facility (RACF) 21, 87
 Response Notification exit 119
 RETAINF parameter, changing in exit 96
 RETAINS parameter, changing in exit 96
 RETRYL parameter, changing in exit 96
 RETRYT parameter, changing in exit 96
 REXX examples, for filters 31
 RFC 1179, LPD command codes supported 36
 roadmap of components to customize 15
 ROUTECDE parameter (JES2) 82
 routing criteria, changing in exit 96
 Routing exit, IP PrintWay
 examples 101
 input 96
 installing 124
 modifying 125
 output 99
 overview 96
 programming considerations 92
 sample 124

S

S2STG DSECT 70
 SAF (system authorization facility) 21
 sample exits 124
 SAP to AFP transform
 converting ABAP data 52
 customizing 43
 environment variable for 49
 mapping bar codes to BCOCA 50
 mapping PJPAPER to AFP page and form definitions 57
 mapping SAP code pages to AFP code pages 59
 mapping SAP OTF fonts to AFP fonts 54
 printing image data 56
 SAP to AFP transform, environment variable for 141
 SCEERUN data set, making it available 18
 scheduler work block text unit (SWBTU) 93
 SCLBDLL data set, making it available 18
 SCS data stream support 10
 SCS data streams, NetSpool exits for 69
 search order for TCPIP.DATA data set 78
 security, establishing 21
 administrator group 22
 aopsetup shell script 22
 operator group 22
 sense codes 143, 144
 separator page, adding in exit 102, 111
 Server Message Block, comparing to other printing protocols 127
 Server Message Block (SMB), customizing
 Print Interface support for 37
 setup string, adding in exit 102
 setup strings for printers
 adding in a filter program 30
 shared printers, SMB 37
 sharing IP PrintWay transmission-queue data set 86
 sharing of printers, customizing PSF for OS/390 for 125
 shutdown procedure, for Infoprint Server daemons 23
 SMB, comparing to other printing protocols 127
 SMB, customizing Print Interface support for 37
 SMF exit, IP PrintWay
 examples 117
 input 115
 installing 124
 modifying 125
 output 116
 overview 115
 programming considerations 92
 sample 124
 SMF record
 replacing in exit 115
 suppressing in exit 115
 SNA sense codes 143, 144
 SNMP agent for OS/390, customizing and starting 61

snmp-community keyword in configuration file 62, 132
 SNMP manager, customizing 63
 SNMP reporting, enabling in Printer Inventory 62
 SNMP subagent
 configuration file statements 62
 customizing 61
 enabling SNMP reporting in Printer Inventory 62
 NPM for the Web, customizing 63
 OS/390 agent, customizing and starting 61
 SNMP manager, customizing 63
 specifying community name for 62
 starting daemon for 62
 SNMP subagent, overview 12
 SPOOLDEF statement (JES2) 80
 start-command keyword in transform configuration file 45, 136
 start-daemons keyword in configuration file 19, 36, 39, 49, 62, 133
 start-directory keyword in transform configuration file 46, 136
 START parameter (JES2) 82
 starting daemons, specifying names in configuration file 19, 36, 39, 49, 62
 startup procedure
 for Infoprint Server daemons 22
 for IP PrintWay 88
 sample 89
 specifying name of in JES2 81
 specifying name of in JES3 83
 for NetSpool 66
 sample 67
 Printer Inventory Manager 22
 SWBTU control block 93
 SYSOUT statement (JES3) 83
 SYSTCPD DD statement 90
 system authorization facility (SAF) 21
 system flow diagram 5
 SYSTEM parameter (JES3) 83

T

TCP/IP
 modifying the PROFILE.TCPIP data set 28
 modifying the TCPIP.DATA data set 77
 specifying the name of 77
 SYSTCPD DD statement in startup procedure 90
 TCPIP.DATA data set 77
 DATASETPREFIX parameter 78
 TCPIPJOBNAME parameter 78
 TCPIPJOBNAME parameter 78
 TCPRCVBUFRSIZE parameter 28
 TCPSENBDFRSIZE parameter 28
 TERM parameter (JES3) 84
 TRACE parameter, in NetSpool startup procedure 68
 TRACEOPT DD statement 68, 69
 transform classes, for PCL to AFP transform 44
 transform configuration file, aopxfile.conf 44

- transform keyword in transform
 - configuration file 45, 135
- Transform Manager
 - configuration file 44
 - customizing 43
 - environment variables for 49
- Transform Manager, overview 7
- transforms, customization required 15
- transforms for OS/390
 - customizing 43
 - dynamic enablement 44
 - transform DLLs 34
- transmission-queue data set, IP PrintWay
 - allocating space 86
 - defining in startup procedure 90
 - initializing 86, 87
 - restricting access 87
 - sharing between IP PrintWay
 - FSSs 79, 86
- Transparent Data Control exit
 - (APIPTD2) 73
- Transparent Data Control exit, NetSpool
 - examples 75
 - input 74
 - installing 76
 - modifying 76
 - output 75
 - programming considerations 69
 - sample 75
- transparent data exits in NetSpool,
 - writing 69
- transparent data support for SCS 10
- TRKCELL parameter
 - JES2 82
 - SPOOLDEF 80
- TRUNC parameter 83
- TSO logon procedure, defining libraries
 - in 24
- TYPE parameter (JES3) 84
- types of LU sessions supported 9

X

- xxxx0000.tab file 59

U

- UNRESTRICTLOWPORTS parameter 28
- URI, for internet printer 39
- URI, used by IPP workstation client 39

V

- VTAM applications, customization
 - required to print from 15

W

- WC parameter (JES3) 85
- web site, downloading Windows client
 - from 127
- Windows client, installing and
 - customizing 127
- Windows client, overview 4
- Windows printing, customization
 - required 15
- WS parameter
 - in DEVICE statement (JES3) 85
 - in PRTnnnnn statement (JES2) 82
- www.ibm.com/printers, downloading
 - Windows client from 127

Readers' Comments — We'd Like to Hear from You

OS/390
Infoprint Server Customization

Publication No. G544-5694-01

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Complete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to find	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to understand	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Well organized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applicable to your tasks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.



Cut or Fold
Along Line

Fold and Tape

Please do not staple

Fold and Tape



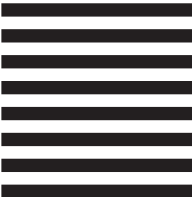
NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
Information Development
IBM Printing Systems Company
Department H7FE Building 003G
Boulder, CO 80301-9817



Fold and Tape

Please do not staple

Fold and Tape

Cut or Fold
Along Line



Program Number: 5647-A01



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

G544-5694-01

